UNIVERSITY OF TARTU
Institute of Computer Science
Software Engineering Curriculum

Muhammad Taimour Khan

Delta VR: Digital Twin

Master's Thesis (30 ECTS)

Supervisor: Ulrich Norbisrath, PhD

Tartu 2025

DeltaVR: Digital Twin

Abstract:

This thesis presents an enhanced version of DeltaVR, building on its predecessor, DeltaVR Multiplayer 2.0. It features a virtual reality-based digital twin of the Delta Centre building, designed to increase interaction within the virtual environment. The platform integrates real-time data visualization through interactive dashboards that display key building metrics, such as energy consumption, CO2 levels, and temperature. The project outlines potential features and experiences that could enhance user engagement and functionality, along with implementation details. Additionally, it explores various methods for presenting and testing the application to different audiences, ensuring its effectiveness across diverse use cases.

Keywords: Virtual Reality, IOT, Digital Twin, Unity, Computer Graphics, User Testing, Game Design, Software Development

CERCS: P170, computer science, numerical analysis, systems, control

DeltaVR: Digitaalne kaksik

Lühikokkuvõte:

See lõputöö esitleb DeltaVR-i täiustatud versiooni, mis tugineb selle eelkäijale DeltaVR Multiplayer 2.0. See sisaldab virtuaalreaalsusel põhinevat Delta keskuse hoone digitaalset kaksikut, mis on loodud virtuaalses keskkonnas suhtlemise suurendamiseks. Platvorm integreerib reaalajas andmete visualiseerimise interaktiivsete juhtpaneelide kaudu, mis kuvavad hoone peamisi näitajaid, nagu energiatarbimine, CO2 tase ja temperatuur. Projekt kirjeldab potentsiaalseid funktsioone ja kogemusi, mis võiksid parandada kasutajate kaasatust ja funktsionaalsust, koos rakendamise üksikasjadega. Lisaks uurib see erinevaid meetodeid rakenduse esitlemiseks ja testimiseks erinevatele sihtrühmadele, tagades selle tõhususe erinevates kasutusjuhtudes.

Võtmesõnad:

Virtuaalreaalsus, IOT, Digitaalne kaksik, Unity, Arvutigraafika, Kasutajatestimine, Mängude disain, Tarkvaraarendus

CERCS: P170, arvutiteadus, arvutusmeetodid, süsteemid, juhtimine (automaatjuhtimisteooria)

Table of Contents

Introduction

The DeltaVR project builds upon three earlier DeltaVR installations (Chapter 2.1). The central problem addressed in this project was integrating real-world data into a digital twin of the Delta building, advancing DeltaVR through Virtual Reality (VR) and Internet of Things (IoT) technologies. Additionally, the project aimed to enhance user experience by introducing innovative features that increase engagement, immersion, and distinctiveness. A further challenge involved effectively presenting the application's features to diverse audiences and collecting feedback. This thesis serves as a resource for improving the DeltaVR project, with each chapter supporting different aspects of its development.

The thesis begins by exploring the background and related works in Chapter 2, followed by a detailed discussion of user experience design. Chapter 3 focuses on the design of the digital twin features. Those looking to implement new experiences or features can refer to Section 3.3, which includes a list of potential enhancements for DeltaVR using emotional goal modeling [7]. This section also provides details on the design and implementation of these features.

Chapter 4 outlines the implementation of the digital twin features, including the dashboard displaying temperature, CO_2 levels, and energy consumption in the building. It also explains how to add a new dashboard for a specific room in the Delta Building by editing the global configuration file. Chapter 5 covers testing and project presentation. Readers seeking guidance on testing the application or presenting its features to an audience will find detailed instructions here, from sending invitations to gathering feedback.

1. Problems

This chapter outlines the issues addressed in this version of DeltaVR, explaining how these problems have been solved and providing some implementation details. A Digital Twin is a virtual representation of a real-world object. Utilizing Digital Twin technology can enhance building management efficiency, leading to improved sustainability and a better overall user experience. In this instance, there's a virtual model of the Delta Centre that uses real-time data to accurately reflect the conditions within the physical building. This virtual model not only depicts the architectural structure of the building but also facilitates dynamic interaction and live monitoring.

The benefits of Digital Twin technology within DeltaVR are significant. For example, real-time monitoring allows building supervisors to oversee important information like temperature, humidity, and weather conditions in the virtual space. Digital Twins can predict potential issues, such as system breakdowns or inefficiencies, which allows for preventive maintenance. This proactive approach reduces both downtime and costs. Additionally, it has the potential to enhance operational efficiency and user satisfaction. Integrating Digital Twin technology into DeltaVR marks a significant advancement in building management and understanding. By combining real-time data, virtual reality, the Internet of Things (IoT), and predictive analytics, DeltaVR can enhance building management and set the stage for future innovations in the industry. This technology has the potential to transform how we monitor, manage, and interact with physical spaces, making them smarter and more efficient in the future.

1.1 Integrating Building Data to Build a Digital Twin

Chapters 4.3 and 4.4 address in depth how building data has been integrated and the technical challenges involved. The challenge centered around enhancing the user experience and optimizing the building's operational efficiency by presenting real-time data in a VR environment. The primary objective was to create a system where users could visualize and interact with various aspects of the building. To achieve this, I utilized digital twin technology and synchronized it with real-world data. The integration required me to extract data from external sources such as the University's InfluxDB[1] and the OIS API[2] using HTTP requests and displaying it within a Unity-based VR application. By leveraging this technology, users could gain insights into various building operations, enhancing their understanding of the building's performance and environment. The goal was to integrate data sources such as weather information and room schedules to ensure that this data was displayed in an intuitive and interactive way within the VR interface.

Integrating building-related data from multiple sources into the *DeltaVR* platform involved handling real-time weather conditions and dynamic room schedules. I retrieved the data in JSON format and processed within Unity to create a user-friendly and visually compelling VR experience. A significant challenge lay in parsing the deeply nested structures of APIs like OpenWeather[1], which provided granular details such as temperature, humidity, wind speed, and specific weather descriptors. Simultaneously, real-time room scheduling data had to be fetched and rendered to reflect accurate availability and usage within the virtual environment.

Effectively displaying weather data in Unity posed another key challenge. The focus was on visualizing various weather conditions—such as temperature, rainfall, and sunlight—within the digital twin. This was accomplished by triggering appropriate game objects, including rain, clouds, or sunlight, based on real-time data retrieved from the weather API. I developed a user-friendly interface (UI) to enable interaction with the weather data, displaying key metrics and allowing users to configure which data points to view. I optimized the UI for VR headsets, featuring clear fonts, large buttons, and intuitive interactions to support an immersive and accessible user experience.

1.2 Enhancing The User Experience & Emotional Goal Modeling

Subchapter 3.3 outlines potential user experiences and features that can be added to enhance the overall user experience. The subchapter also provides details on their implementation in Unity and examines the emotions they are intended to evoke. Identifying the experiences involved using emotional goal modeling, which is identifying the desired emotional responses from users during their VR experience, such as excitement, empathy, or joy, and designing scenarios to evoke those emotions effectively. For example, a maze experience within the Delta Centre building could create a sense of thrill and adventure for the user. Identifying experiences also required a deep understanding of what makes VR experiences compelling, as well as evaluating

[1] https://openweathermap.org/

the technical feasibility of integrating these new features into DeltaVR.

Adding these new experiences offers several benefits. It increases user engagement, makes the platform stand out from competitors, enhances immersion, and improves overall user satisfaction. Interactive and immersive elements, such as real-time weather changes and building environment interactions, help users feel more connected to the virtual space. This can lead to positive feedback, longer usage times, and a growing community of users. Additionally, adding new features provides an opportunity to experiment with cutting-edge VR technologies, ensuring that DeltaVR remains innovative and aligned with industry trends.

There are several reasons why I used emotional goal modeling to identify the new experiences and features. Firstly, it helps create a more engaging and memorable experience for users. When users feel a specific emotion, they are more likely to form a strong connection with the content, leading to a deeper level of immersion. For example, experiences designed to evoke empathy can enhance users' understanding of the building's environment and the individuals interacting with it. By creating an emotional connection, these experiences encourage a sense of responsibility toward sustainable practices and thoughtful resource management.

Emotional goal modeling enhances user retention and satisfaction by creating emotionally engaging experiences that encourage users to revisit the platform and share their positive impressions, fostering community growth. By designing features that evoke emotions such as excitement or empathy, Delta VR can differentiate itself from entertainment-focused VR applications. This approach entertains users and inspires meaningful interactions and thoughtful engagement, enhancing DeltaVR's impact as a tool for building management, operational insights, and immersive virtual experiences.

The process of coming up with the new experiences and features involved researching existing VR applications, such as VRChat[2], Rec Room[3], and Resonite[4], as these are among the most popular social VR platforms offering rich interactive experiences. By exploring these

[2] https://hello.vrchat.com/
[3] https://recroom.com/
[4] https://resonite.com/

applications through methods like immersion, usability analysis, relevance to DeltaVR, and technical feasibility for integration with DeltaVR, the project aimed to gain insights into what engages VR users and how similar experiences could be adapted or reimagined for DeltaVR. This involved analyzing popular features, such as customizable avatars, social spaces, interactive mini-games, and collaborative environments.

Another challenge involved in the process was understanding DeltaVR’s audience, which includes VR and IoT enthusiasts, professionals seeking immersive tools, current and prospective students of the University of Tartu, and individuals interested in virtual tours of the building. This audience likely differs significantly from the typical users of entertainment-focused VR platforms. DeltaVR users may prioritize practical functionality and operational insights over purely entertainment-driven features.

It was crucial to balance useful real-world features with immersive and enjoyable experiences. The goal was to enhance user engagement without deviating from DeltaVR's core purpose of providing building-related data and operational insights. The implementation details were another critical aspect of the problem. After identifying potential new experiences, it was necessary to evaluate how they could be effectively integrated into DeltaVR.

1.3 Presentation of DeltaVR

The DeltaVR project incorporates a process for presenting and testing the application with diverse audiences, as outlined in Section 5.2 of the thesis. This process includes preparing and installing the application, identifying and inviting target audiences, selecting suitable presentation venues, collecting user feedback, and addressing potential challenges during testing. Each step ensures the app is effectively demonstrated, tested, and improved based on user input.

The process begins with installing the DeltaVR APK on Meta Quest devices, requiring technical setup through the Meta Quest Developer Hub[5]. This step poses logistical challenges, such as

[5] https://developers.meta.com/horizon/downloads/package/oculus-developer-hub-win/

ensuring compatibility across devices and troubleshooting installation issues. The project establishes a smooth technical foundation for presentations by tackling these challenges.

Identifying the target audience involves understanding the application's purpose and appeal, which ranges from VR enthusiasts and IoT developers to students and educators. This diversity necessitates tailoring the presentation to align with the audience's varying technical backgrounds, interests, and expectations. Engagement strategies include leveraging social media, academic connections, and online communities to invite participants, ensuring the event reaches relevant individuals.

Feedback collection is central to refining DeltaVR. The process involves recording user interactions, noting areas of confusion, and gathering suggestions for improvement. Feedback methods range from surveys and interviews to informal discussions, ensuring insights are gathered effectively. Challenges in this stage include managing diverse feedback from audiences with varying expertise levels and prioritizing actionable suggestions.

Presentations are designed to accommodate both individuals and groups. The process includes familiarizing users with VR controls and observing their experiences to address obstacles in real-time. For example, participants unfamiliar with VR are guided through a live demo, while experienced users receive verbal explanations. Recording feedback is methodical, with testers' information and responses organized under separate headings to create detailed personas. Potential challenges, such as technical issues during presentations or varying engagement levels, are proactively addressed by ensuring a backup plan and adapting the presentation to audience interests.

Overall, the presentation and testing process is vital for aligning DeltaVR with user needs. It enhances the application’s usability, identifies areas for technical and experiential improvements, and fosters a deeper understanding of user expectations. By iteratively refining the application through structured feedback, the process not only optimizes DeltaVR’s functionality but also strengthens its appeal to a broad and diverse user base.

2. Background

This chapter provides context and background for the development of the prior work that has been completed before this project. Subchapter 2.1 describes the previous DeltaVR installations that have been developed. Subchapter 2.2 explains the elements that make up the design of VR experiences and their impact on user perception and engagement, as these elements help to come up with potential experiences that can be added in DeltaVR.

Subchapter 2.3 explores similar projects, particularly focused on creating digital twins. Chapter 2.3.1 contains details about the visual representation of a university campus, which is similar to DeltaVR. Chapter 2.3.2 summarizes a study that investigates the integration of digital twin technology with a building's Heating, Ventilation, and Air Conditioning (HVAC) system, which enables continuous monitoring and analysis, providing valuable insights to optimize energy efficiency and building performance. Together, these sections establish a foundation for understanding the key principles and inspirations driving the DeltaVR project, offering insights into its objectives and the innovations it seeks to achieve.

2.1 Previous Delta VR Installations

DeltaVR, developed as part of a bachelor's thesis (2021), is a VR experience showcasing the Delta educational building, built from the DBV[6] project's 3D model after optimizing frame time and fixing graphical bugs [10]. The experience includes spatial movement, virtual hands, interactive doors, and games like archery and Breakout VR. User testing praised its rendering quality and interactivity, though feedback highlighted room for improvement in controls and adding features. The project successfully minimized cybersickness and set a strong foundation for future enhancements.

The DeltaVR Multiplayer [11] project built upon its predecessor to enhance functionality, usability, and performance. Key advancements included expanding the virtual Delta Building to

[6] https://dbv-sc.weebly.com/

all three floors, refining existing experiences like the archery range, and introducing new features such as a drawing experience and a virtual spacewalk.

Usability and performance testing highlighted improvements in reducing gameplay breakdowns, eliminating the need for an external internet connection, and adding a tutorial to ease navigation. Despite these enhancements, minor usability challenges, bugs, and slight nausea issues persist, with some attributed to headset fit. Overall, the project delivered a more immersive and stable VR experience while identifying areas for further optimization.

The DeltaVR Multiplayer 2.0 [1] project achieved significant enhancements to the original application, addressing critical bugs and introducing various improvements to the virtual Delta Building and user experiences. The development expanded the virtual environment to include all three floors of the Delta Building, compared to the previous version's limitation to the second floor. This expansion, alongside optimizations such as light baking and improved frame times, resulted in greater visual fidelity and performance stability.

Usability testers praised the realistic detail of the virtual building, and performance testing demonstrated reduced desynchronization issues, enhancing overall gameplay stability. A key advancement was the inclusion of a networking solution that removed the need for an external internet connection, streamlining the application for demonstrations. Additionally, the introduction of a tutorial improved user navigation and reduced reliance on external assistance.

New features were added, including a refined archery range and two new experiences: a drawing experience and a virtual spacewalk, which received positive feedback despite minor usability challenges. Efforts to minimize motion sickness showed progress. Usability and performance testing throughout the project provided valuable insights into further optimization needs. In conclusion, DeltaVR Multiplayer 2.0 succeeded in delivering a more immersive, detailed, and stable VR experience while laying the foundation for future improvements and innovations.

Up till now, the DeltaVR installations project has developed a detailed 3D model, offering opportunities to integrate building data for enhanced functionality. This thesis takes advantage of the opportunity to convert the project into a digital twin. The current thesis further improves the previous work by adding a detailed guide on presenting the application to different audiences. There have been experiences included in the DeltaVR installations, and the current project outlines potential new experiences along with their implementation details, paving the way for further expansions and innovations.

2.2 Elements of Virtual Reality UX Design

This chapter explains the elements of UX design that have to be considered while developing a VR experience. The elements include immersion, sense of control, pleasantness, exploration, disorientation, presence, and agency. These elements were used along with emotional goal modeling to analyze different experiences and then list the ones that most likely can be added to this version of Delta VR (Chapter 3.3.1) to make the overall experience more interesting for the users.

In their exploration of virtual reality design, Kauhanen et al. [5] analyze key experiential elements crucial for creating immersive and engaging VR environments. The most important part of the virtual reality experience design is the concept of immersion. It refers to the level of sensory fidelity provided by a VR system. Jerald [6] further refines the concept of immersion in "The VR book", explaining that matching ensures sensory congruence, surroundness creates a panoramic experience, vividness enhances simulation quality, interactability enables user influence, and plot maintains narrative consistency.

Presence, closely tied to immersion, is the user's subjective feeling of "being there" in the virtual environment. It is influenced by the immersive quality of the technology and encompasses multiple dimensions and subcategories. However, the multitude of definitions for presence can lead to confusion and hinder understanding of the phenomenon.

Disorientation is another significant aspect in VR experiences, particularly in large and complex virtual environments. It entails a feeling of loss of location and can cause frustration and disengagement among users. Understanding and addressing disorientation is crucial for enhancing user satisfaction and engagement. A sense of control or agency is essential for a positive VR experience, as users should feel that their actions have meaningful consequences within the virtual world. Reduced sense of control can lead to frustration and dissatisfaction among users when their interactions do not produce expected outcomes.

It's important to recognize the nature of agency in VR experiences. In VR, spatial-explorative agency lets players physically navigate virtual worlds using their avatar, enhancing immersion by mimicking real-life movement. VR technology, including headsets and motion controllers, intensifies this experience, providing a more immersive way to explore digital spaces. Temporal-ergodic agency involves interacting with the game world over time, influencing events, and witnessing dynamic changes. VR environments can simulate realistic day-night cycles and weather patterns, allowing players to experience a sense of temporal presence as they navigate through evolving virtual worlds. Configurative-constructive agency empowers players to customize avatars and shape the game world in VR. Platforms offer extensive customization tools and creative building features, enabling players to express themselves and exert control over their virtual environment.

Narrative-dramatic agency allows players to actively participate in the game's story and influence its unfolding narrative through their avatar's actions and choices. VR storytelling techniques engage players in emotionally compelling narratives, making them active participants in the unfolding drama. By differentiating between spatial-explorative, temporal-ergodic, configurative-constructive, and narrative-dramatic agency, we gain valuable insights into how these dimensions interact and shape an experience. While our focus may lean towards narrative-dramatic agency [5], it's crucial to acknowledge the relationship between all four dimensions.

Pleasantness, characterized by positive emotional valence, is another key factor in VR design. VR content should evoke satisfaction, joyfulness, relaxation, and fascination to create enjoyable experiences for users [5]. Exploration plays a significant role in VR content design, especially for environments like museums and cathedrals. VR offers the unique opportunity for users to immerse themselves in new worlds and environments, facilitating enhanced learning experiences through exploration. In the case of the development of one of the most played VR games, Half-Life: Alyx[7], an iterative design methodology proved effective in leveraging player exploration. Valve tapped into how players naturally want to explore in VR. During testing, they noticed players were spending way more time checking out every nook and cranny of each room, which isn't typical in regular PC games.

In VR, people just want to take their time, look at everything, and soak in the environment. So, Valve smartly focused on making those exploration moments even more exciting. They poured extra effort into the details and added interesting stuff to discover, based on what players were loving the most. This careful crafting, driven by player feedback, made exploring in Half-Life: Alyx a truly immersive and rewarding experience from start to finish. Finally, simulator sickness is an important consideration in VR design, as it can lead to physical discomfort and affect user experience. It encompasses symptoms such as nausea, difficulty focusing, and vertigo, often stemming from visual display and visuo-vestibular interaction.

[7] https://store.steampowered.com/app/546560/HalfLife_Alyx/

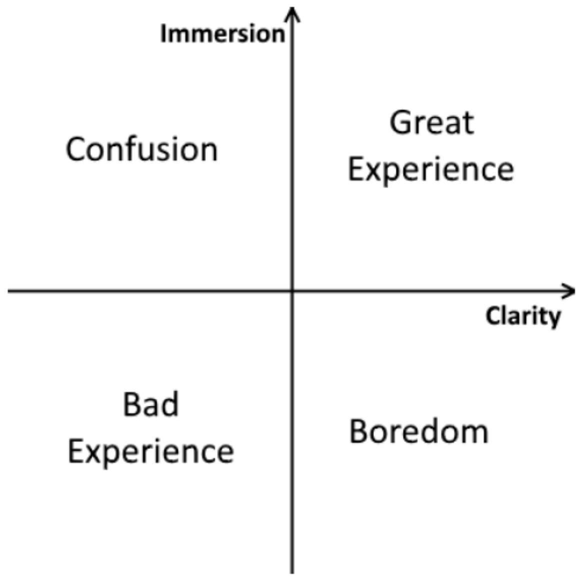


Figure 1. A measurement tool that assesses immersion and clarity in various game elements

The tool shown in Figure 1 can facilitate the analysis of design decisions, problems, and solutions throughout a project [3]. By objectively evaluating the level of immersion and clarity in each aspect of the game, the tool provides valuable insights into player experience. It enables designers to identify elements that may cause confusion or disengagement. The focus of the analysis lies in VR environments, acknowledging that strategies proposed may not apply to non-VR games. And through this approach, the overall quality of virtual reality gaming experiences can be enhanced.

2.3 Similar Digital Twin Projects

This section provides an overview of other projects that have implemented digital twin technology in educational and building management contexts. By examining these examples, it becomes possible to understand how digital twins are applied across different domains and how similar approaches have informed aspects of the DeltaVR project. These case studies offer practical insights into the use of virtual environments for learning, campus engagement, and real-time monitoring of physical spaces.

2.3.1 Metaversity

Digital Twin Metaversities[8] are advanced digital replicas of physical educational institutions, including campuses, colleges, and universities. These virtual environments are crafted by a team of 3D modelers, developers, and designers to provide an immersive and realistic experience of a university setting within the metaverse. In addition to replicating existing campuses, the team also creates entirely new virtual spaces that can't be built in the real world, offering unique opportunities for innovation in education and beyond. These digital twin metaversities enable students, faculty, and other users to explore and interact with their campus grounds in a virtual reality environment. The hyperrealistic nature of these spaces ensures that the virtual experience closely mirrors the in-person experience, making it a valuable tool for various applications.

Once developed, these digital twin campuses can be used for instruction, hosting events, running marketing campaigns, recruiting new students, and more. The team has partnered with over 50 higher education institutions globally to bring their campus replicas into the metaverse, showcasing a growing trend towards integrating digital twin technology in education. Additionally, VXRLabs[9] (VXRL) offers a fully immersive platform that hosts a wide range of educational experiences and shared spaces, catering to subjects like sciences, career and technical training, healthcare and medical simulation, and history. This platform further enriches the educational landscape by providing content that can be used individually or in multiplayer formats, enhancing both learning and collaboration in a virtual setting.

2.3.2 Digital Twin Concept Study

Sanchis et al [9]. presented a study about using digital twin technology to enhance the efficiency and sustainability of buildings on a university campus. A digital twin is essentially a virtual replica of a physical space that monitors and optimizes various environmental aspects in real-time. The researchers focused on an open floor-plan office space, employing sensors and cameras to track conditions such as temperature, humidity, and occupancy. By digitally

[8] https://www.victoryxr.com/metaversity/
[9] https://www.victoryxr.com/vxrlabs/

reconstructing the physical space, the team could accurately represent real-world data in the virtual environment. Ambient sensors were deployed to collect data on temperature and humidity, which was then fed into the digital twin to provide a real-time view of the space's conditions. To detect occupancy, the system used a single camera paired with a computer vision algorithm to determine how many people were in the room and their locations. This information is crucial for optimizing energy usage, such as adjusting heating or cooling based on room occupancy. The data gathered from the sensors and camera were streamed in real-time to the digital twin using an Internet of Things platform called ThingSpeak, ensuring that the virtual model stayed updated with the latest information. This integration allowed the digital twin to stay synchronized with the physical space, continuously incorporating sensor data and occupancy information. Additionally, the digital twin provided visual representations of real-world phenomena, like airflow from the heating, ventilation, and air conditioning (HVAC) system, helping users to better understand how the space was being managed.

The study also identified several challenges in implementing digital twins, including difficulties in collecting accurate data, the significant computational resources required, and security concerns. Despite these challenges, the study indicated plans to add more features, such as lighting and HVAC control, to make the digital twin even more comprehensive. This research is highly relevant to Delta VR, as it demonstrates how digital twins can be used to monitor and optimize physical spaces. DeltaVR is already highly immersive, excelling in all key UX design elements. However, the UX and specifically player agencies (described in Chapter 2.2) can be further enhanced by implementing the potential experiences outlined in Chapter 3.3. Similar projects mentioned in this chapter, specifically subchapter 2.3, have informed and improved the current DeltaVR project, ensuring a refined outcome. Additionally, energy efficiency in the Delta Building can ultimately be optimized, as demonstrated in the study discussed in Chapter 2.3.2, presenting a valuable opportunity given the availability of both a VR platform and relevant building data.

3. Design

This chapter contains the design of all the features of the project. Subchapter 3.1 explains the design of the weather integration. Subchapter 3.2 describes the design of the building data integration (CO_2, temperature, energy, and schedule data). Subchapter 3.3 presents the ideas and features that can be added to the current deltaVR experience to make it more interesting to the users, ranked according to their suitability. This subchapter also contains technical details and design that will help anyone wanting to enhance the project by adding experiences and features.

3.1 Weather API Integration

In this project, the first step to convert the DeltaVR project into a digital twin by connecting it to the physical world is to enable users to explore real-time weather conditions. The weather should be shown on a dashboard. It can be retrieved with the press of a button. The dashboard will show different icons for different weather conditions, e.g, if it's cloudy, then an icon representing a cloud will be shown. The values for the minimum and maximum temperature, wind speed, pressure, humidity, sea level, and ground level are to be shown.

3.2 Building Data Integration

The second and major step in transitioning to a digital twin is to connect the building data and classroom schedules to the virtual model. The goal is to provide an immersive and interactive experience for users to understand the present CO_2 levels, energy consumed, temperature, and schedules of different rooms inside the building. Interactive panels are to be placed throughout the Delta building, allowing users to engage with real-time data. The data should be updated every 2 minutes. There should be buttons for switching between the different metrics. It must ensure that users can easily explore and understand the building's performance metrics within the VR environment. Boards, along with the schedule information of a classroom, are present right outside the respective room in the Delta building. The aim is to duplicate it in the digital twin and display the schedule of a room outside on board. The board in the digital twin must show the accurate room number, the course being taught, and the teacher responsible for different classrooms.

3.3 Ideas and Features for Enhancing the Experience

This chapter outlines potential features and experiences that can be integrated into the DeltaVR application to enrich the overall user experience. During the thesis, I analyzed various VR experiences and games which include VRChat and Resonite, using the criteria of immersion, presence, disorientation, sense of control or agency, pleasantness, and exploration [5] to identify those best suited for DeltaVR. I surveyed the most popular VR experiences and. High sensory fidelity, user-friendly navigation, dynamic environments, and personalized features were prioritized. These elements ensure DeltaVR delivers immersive, creative, and emotionally engaging experiences that cater to diverse user preferences in the future. The chapter consists of eight potential experiences and four features. Each subchapter contains some basic information about the experience, what it will be like in the context of DeltaVR, and its benefits. For the experiences, the emotions they will create in the user are also present. The implementation details of each experience and feature are present as well.

3.3.1 Visual Scripting

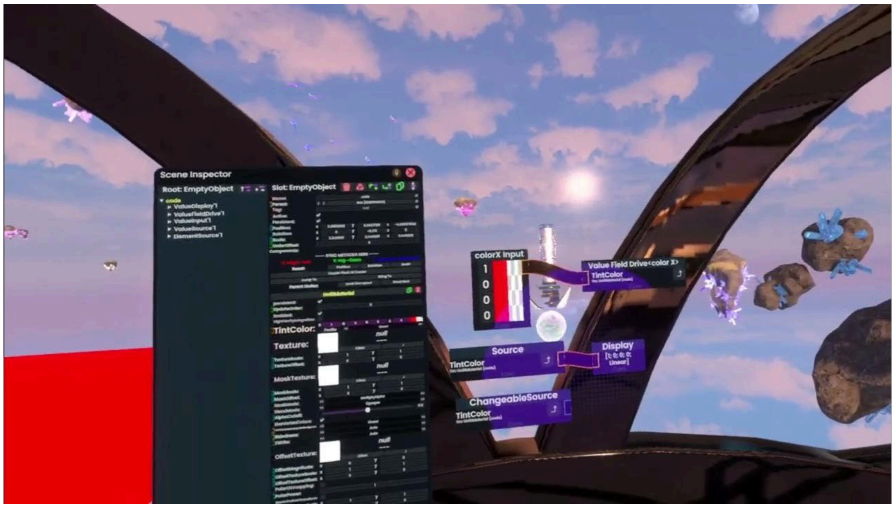


Figure 2. Creating a feature/animation through nodes in Protoflux

A visual scripting tool can be added to DeltaVR which can let users create interactive and dynamic experiences without writing code. It would work by using nodes, which are small building blocks that control different actions, objects, and effects in the virtual world. Users may spawn, connect, and modify nodes with the Tool, making it easy to build custom interactions. It would also support different types of variables for storing and sharing data and offers a wide range of node categories like audio, avatars, actions, and objects. This would make it a powerful tool for customizing VR experiences and adding interactivity.

In DeltaVR, users would be able to create and customize their own interactive experiences within the virtual Delta Building. This could include custom minigames, automation of in-game events, and enhanced user interactions. A node-based scripting system could allow users to visually connect different functions without needing programming knowledge, making DeltaVR more dynamic and customizable.

To use this system, players could select and place different functional blocks from a menu, then connect them to define actions and behaviors. These blocks may control objects, sounds, animations, and more, enabling users to build interactive experiences directly within the VR environment. Once placed, users may easily adjust and link them using a laser-like tool, making the process intuitive. However, since this system has many possibilities, tutorials may be needed to help users fully unlock its potential. By allowing players to shape their own experiences, this feature could significantly enhance creativity and engagement within DeltaVR.

This experience may create a range of emotions in users. Using it initially and frequent interaction with it would likely induce a sense of curiosity and excitement as users explore the creation of dynamic experiences. As they start to understand and utilize the tool, feelings of empowerment and creativity may emerge, driven by the ability to bring their ideas to life within the virtual environment. Successfully connecting and scripting nodes could give users a sense of accomplishment and satisfaction. It may also create a strong sense of engagement and fulfillment as users become more used to it and see their unique concepts materialize within DeltaVR.

While Unity offers a robust development environment with tools like the hierarchy, inspector, and scene views along with scripting through C#, its system often demands programming knowledge and familiarity with software development concepts. This can create a barrier for users who are not technically inclined.

In contrast, the visual scripting system proposed for DeltaVR simplifies the creation process by removing the need for traditional coding. Instead of writing scripts, users can build functionality by connecting visual blocks that represent different actions and behaviors. This approach is more intuitive and accessible, especially in a VR setting, where users can interact directly with elements using tools like virtual pointers or laser controls. Compared to Unity's conventional workflow, visual scripting would make DeltaVR more inclusive, enabling a wider range of users to participate in content creation and improving creativity and engagement within the platform.

This experience can be implemented using the following steps:

1. VR Scripting Environment: Develop a VR-compatible scripting interface that allows users to create, test, and deploy dynamic experiences.
2. Integration with DeltaVR: Ensure seamless integration with existing DeltaVR systems and assets.
3. User Interface: Design a user-friendly UI/UX for scripting, including tools for debugging and visualization within VR.

4. Documentation: Provide documentation and tutorials to help users learn and utilize the scripting tool.

3.3.2 Avatar corridor

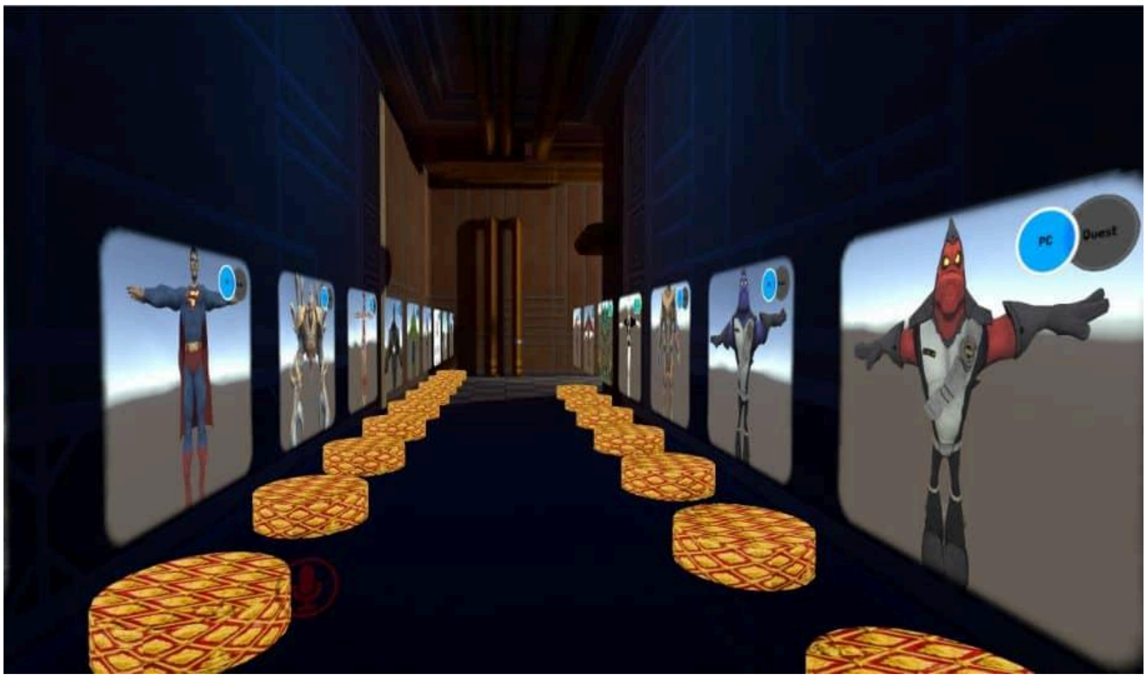


Figure 3. Example of an Avatar Corridor

Avatar Corridor is a proposed feature in DeltaVR designed to provide users with an engaging and immersive way to select and change their avatars. This feature would likely enhance user experience by giving them control over how they appear while exploring the Delta building. Any suitable part of the DeltaVR environment could be adapted into an avatar corridor, offering flexibility in placement. With a wide variety of characters to choose from—primarily cartoon-style avatars but with potential for more realistic options—users might feel excitement and curiosity as they explore.

Familiar characters could spark nostalgia and enjoyment, while the corridor's immersive design may contribute to a sense of discovery and adventure. However, the visual style of the corridor could benefit from aesthetic improvements, and expanding the variety of avatar styles beyond cartoons might make the experience more inclusive and appealing.

Users may enter a designated area of the Delta building that functions as an "avatar selection corridor." This space could resemble an underground tunnel or a futuristic hallway to heighten the immersive experience. Avatars can be displayed on either side of the corridor, similar to a gallery setup, with each avatar positioned on a pedestal or inside a display case. Clear labels may indicate the theme or origin of each avatar (e.g., animated shows or fictional universes).

Users could approach an avatar and interact with it—through a VR gesture like pointing, touching, or clicking—to try it on. Once they find an avatar they like, they may have the option to select and save it as a favorite for quick access in future sessions. Navigation through the corridor can be supported by interactive panels or buttons labeled “Next” or “Previous,” allowing users to browse through different avatar sections with ease. Signage and visual cues may guide users intuitively through the space.

This experience may be implemented using the following steps:

1. Design Environment: Create a visually appealing avatar corridor with clear signs and pathways.
2. Avatar Integration: Curate a diverse collection of avatars from popular shows and realistic themes.
3. Interaction Mechanism: Enable touch or click interactions to let users try on and preview avatars.
4. Save Feature: Add a "Save to Favorites" option for easy access to selected avatars later.
5. Navigation System: Implement “Next” and “Previous” buttons or filters for smooth browsing.
6. Emotion Enhancement: Use themed music, sound effects, and hidden avatars to evoke nostalgia and excitement.
7. Continuous Updates: Regularly rotate avatars, improve aesthetics, and address user feedback.

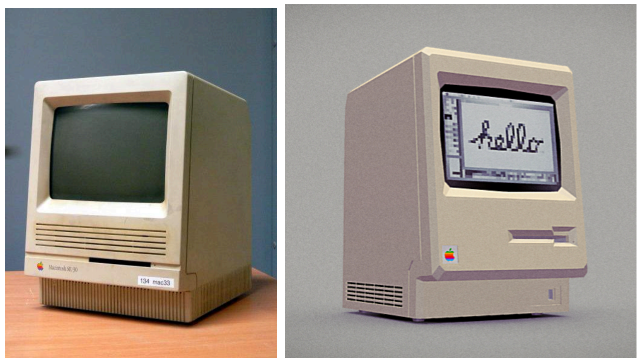


Figure 4. Left: A real-life Macintosh present in the Computer Museum. Right: A 3D Model of the Macintosh.

3.3.3 Computer Museum

The computer museum is an essential part of Delta Center, showcasing the evolution of computers and technology. It holds valuable historical information, allowing visitors to explore advancements in computing over time. By presenting this content interactively and engagingly, the museum serves as an educational resource, making learning about technology more immersive. To enhance the Delta VR experience, incorporating the computer museum and its components will make the virtual tour more interesting, educational, and comprehensive.

Incorporating the computer museum into the DeltaVR experience is likely to evoke a range of emotions, enhancing user engagement and enjoyment. Users can experience curiosity and interest as they explore detailed 3D models and historical information about the evolution of computers. Nostalgia can be evoked for those familiar with older computer models, while the immersive and interactive nature of the VR experience will inspire awe and wonder. Educational satisfaction will arise from learning about the history and impact of various computer models in an engaging way, keeping users engaged and excited through interactive features like clickable models and animations. Users affiliated with Delta University may feel a sense of pride and connection seeing their institution's contributions highlighted. Lastly, discovering lesser-known facts about the evolution of computers can delight and intrigue users. By combining education with interactive and immersive experiences, the computer museum in DeltaVR can inform and emotionally engage users, making their virtual tour memorable and impactful.

The DeltaVR experience can integrate the computer museum by allowing users to navigate through a virtual exhibition featuring 3D models, animations, and interactive elements. Users could click on models, view animations, and learn about historical milestones in computing in an engaging and immersive way. By blending education with interactivity, the virtual museum can provide a dynamic and memorable learning experience within DeltaVR.

This experience may be implemented using the following steps:

1. Design Virtual Space: Create a 3D exhibition area for the computer museum in DeltaVR.
2. 3D Model Integration: Add detailed models of historical computers and technological milestones.
3. Interactive Features: Implement clickable models and animations to showcase advancements in computing.
4. Educational Content: Display key historical information alongside each model for an immersive learning experience.
5. Emotional Engagement: Use nostalgic designs and storytelling to evoke curiosity, wonder, and pride.
6. Navigation System: Develop clear pathways and signs for easy exploration of the virtual museum.

3.3.4 Virtual Drawing

Figure 5. A 3D model made through virtual drawing

Another potential feature for DeltaVR is Virtual Drawing, an experience where users could draw and model in a 3D virtual space. This concept was proposed due to its potential for future development, specifically, a VR-to-3D-printing extension that may allow users to create models in VR and later print them as physical objects. The core of the experience focuses on creativity and collaboration, enabling users to draw freely using various strokes, lines, and shapes to bring their ideas to life in a shared 3D environment.

The collaborative nature of the feature could encourage users to interact, co-create, and learn from one another. Beyond being a fun and novel activity, Virtual Drawing may also serve an educational purpose, introducing users to basic principles of 3D modeling, a skill relevant in design, engineering, and animation. As a result, the feature can contribute to DeltaVR's value as both an interactive and educational platform.

Emotionally, users may experience a strong sense of creativity, curiosity, and inspiration while experimenting in the 3D space. The ability to visualize and potentially print their creations could lead to feelings of accomplishment and satisfaction. Exploring the tools, collaborating with others, and seeing ideas take shape might make the experience deeply engaging and enjoyable.

Users of all ages could find it appealing, particularly due to its playful, hands-on nature and the opportunity to be part of a creative, forward-thinking community.

This experience can be implemented using the following steps:

1. Design Virtual Drawing Space: Create an intuitive 3D environment for users to draw and model.
2. Drawing Tools Integration: Add various brushes, strokes, and shapes to enable creative freedom.
3. Export Feature: Develop functionality to export user-created 3D models.
4. Collaboration Mode: Enable multi-user sessions for shared drawing and collaborative projects.
5. Educational Features: Introduce guided tutorials for learning 3D modeling basics.
6. Interactive UI: Implement easy-to-use controls and toolkits for seamless creativity.
7. User Engagement: Incorporate social sharing, teamwork challenges, and project showcases.

3.3.5 District Roboto

This experience is composed of neon-lit streets of the far future, where humanoid robots go about their daily lives, all with stunning graphics and lighting. This experience could be another one thematic to the robotics lab present in the building, along with making players aware of how much VR experiences have progressed, as it is among the most visually impressive concepts proposed. The environment and lighting can be highly detailed, and the robots could be well animated. Some features, such as eating food and drink, may not add much utility in the absence of player health or status systems. There could also be elements like food stalls which appear interactive but currently are not—offering room for improvement in interactivity.

Players in DeltaVR can have the option to open a navigation menu from which they can choose to start this experience. The experience may allow players to roam around and discover a world of robots. There could be a wide variety of robots present. Players can mingle with the robots on the street and eavesdrop on their conversations, pop into a restaurant for a drink or some food, or interact with one of the stray cats that roam around. There may be several streets to explore, with the option to climb to different areas.

This futuristic, neon-lit cityscape in DeltaVR, populated by humanoid robots, can evoke different emotions in players. The stunning graphics and detailed lighting could inspire wonder and amazement, captivating players with its visually immersive environment. Curiosity may drive players to explore every corner of the city, interacting with the diverse robots and observing their behaviors.

The immersive nature of the environment, with detailed animations and potential interactive elements, can deeply engage players, whether they are visiting virtual restaurants or interacting with roaming cats. The ability to mingle with robots and explore the city could create a strong sense of presence and interaction, fostering a deeper connection to the virtual world. The experience may also offer an escape into a fantastical realm of advanced technology, enhancing enjoyment and providing a break from reality. Additionally, exploring alongside other players can foster a sense of community and shared experience, adding layers of social interaction and connection to the overall experience.

This experience can be implemented using the following steps:

1. Design Neon-Lit Cityscape: Create a detailed futuristic environment with stunning lighting, vibrant streets, and humanoid robots.
2. Develop Robot Animations: Ensure diverse, lifelike robot behaviors, conversations, and routines for an immersive experience.
3. Add Interactive Elements: Implement interactive food stalls, restaurants, and stray animals like cats to enhance engagement.
4. Enable Exploration Features: Allow players to roam freely, climb structures, and explore multiple streets.
5. Navigation Menu Integration: Add a user-friendly navigation menu in DeltaVR to start and manage the District Roboto experience.
6. Foster Social Interaction: Enable multiplayer exploration and mingling for shared experiences and connection-building.

7. Optimize Graphics & Lighting: Use high-quality graphics and dynamic lighting to maximize visual immersion and player awe.

3.3.6 Tag

This experience is a VR-based tag in which two teams play tag in a maze-like environment. It is a multiplayer experience. Which can be placed inside DeltaVR to make the experience more enjoyable and immersive. It’s a collaborative experience and involves communication. The maze environment makes things very interesting, and the overall experience is thrilling. This idea is taken from VRChat, and if the graphics of the original experience can be improved, it would make it more enjoyable for the user. The Freeze Tag experience inside the Delta Building of DeltaVR will transform familiar spaces into an interactive, adrenaline-fueled adventure.

The game takes place in a maze-like environment within the Delta Building, designed to offer both strategy and surprise at every turn. Players will be divided into two roles: taggers and runners. The goal for taggers is to capture all runners before the timer runs out, while runners must avoid being caught and unfreeze teammates to secure a win. As players navigate the intricate hallways and rooms of the Delta Building, the maze will constantly change. Gates will randomly appear and disappear, blocking pathways or opening unexpected escape routes. Portals placed around the map will offer instant teleportation, providing quick exits or shortcuts to help runners evade taggers or for taggers to close in on their prey.

The dynamic environment ensures that each round is unpredictable and filled with suspense. For runners, the thrill of evading taggers will create moments of high energy, especially as they race to unfreeze teammates before being caught themselves. The strategic placement of speed-boosting crystals throughout the Delta Building will provide bursts of speed, giving players a temporary sense of invincibility and control over the game’s pace. Obstacles like rotating stones that fire lasers will introduce additional challenges, forcing players to stay alert and adjust their strategies on the fly.

Taggers will feel a strong sense of determination as they close in on their targets, and capturing a runner will bring immense satisfaction, especially if it shifts the momentum of the game. Runners, on the other hand, will experience relief and triumph when narrowly escaping or successfully freeing a teammate under pressure. The teamwork dynamic will be crucial to the Freeze Tag experience. Runners will need to coordinate with each other to stay out of danger and rescue frozen teammates, fostering camaraderie and a deeper sense of connection. The constant action and evolving strategies will inspire playful competition and joy, keeping players engaged and eager to return for more. Overall, the Freeze Tag experience within the Delta Building will combine fast-paced action with social interaction, creating a thrilling, dynamic, and unforgettable game environment in DeltaVR.

These are the steps to implement this experience:

1. Design Dynamic Maze Environment: Create maze-like hallways and rooms within the Delta Building, featuring randomly appearing gates and teleportation portals for unpredictable gameplay.
2. Define Player Roles & Objectives: Set up clear roles for taggers and runners, with taggers capturing runners and runners unfreezing teammates to win.
3. Add Interactive Elements: Incorporate speed-boosting crystals and laser-firing obstacles to enhance gameplay challenge and provide tactical advantages.
4. Enable Team Coordination: Implement communication tools or indicators to help runners coordinate escapes and rescues.
5. Ensure Dynamic Round Variation: Randomize maze layouts, gate placement, and portal locations each round to keep the game fresh and exciting.
6. Provide Feedback & Rewards: Offer visual or audio cues for key actions (e.g., unfreezing teammates or capturing runners) to enhance engagement and player satisfaction.
7. Optimize Graphics & Immersion: Improve the visual design and lighting to heighten immersion and ensure a thrilling, fast-paced experience for all players.

3.3.7 Jackal Hunt

This experience involves a game in the Delta building in which players could face enemy robots and fight for survival. As the game begins, the player may find themselves standing in the lobby of the Robotics Lab at Delta University. The room could be dimly lit, with shadows moving across the walls, creating an atmosphere of suspense. With a sense of determination, the player may set out to locate the corrupted jackals that may threaten the lab's environment. As they move through the lobby, they could scan the surroundings, searching for any signs of movement. A metallic growl may echo through the room, alerting the player to the presence of their first adversary. They could draw their water gun and aim. The robotic jackal might emerge from the shadows, its glowing circuits pulsating with energy. The player may squeeze the trigger, sending a stream of water towards their target. The health of the jackal could decrease, which might be shown through a bar.

After sufficient hits, the jackal's circuits may short-circuit, causing it to collapse in a heap of metal and sparks. As the player continues their hunt, they may consult the map displayed on their HUD, which could be dotted with red markers indicating the locations of nearby jackals. Each successful takedown may add to their score, displayed prominently on the screen alongside the number of jackals remaining. Meanwhile, a gauge on the side of the screen could indicate the level of water in the player’s gun, reminding them to conserve their ammunition. In the corner of their vision, a real-time display may show the current CO_2 levels in the lab, steadily rising due to the corrupted jackals' presence.

With every jackal defeated, the player could feel a sense of satisfaction and accomplishment, as they may be making a difference in restoring balance to the lab's environment. As they press onward, navigating through the corridors of the building, they may be filled with a sense of purpose, ready to face the challenges that lie ahead. But this game could be more than just a test of skill—it may be an opportunity to explore and learn. As the player navigates through the virtual Delta Centre, they might encounter various rooms, labs, and inhabitants. The spatial aspect of the game could serve as a way to understand the layout and purpose of the building, making the experience both educational and engaging.

This immersive VR experience may build a deeper connection to Delta University and its cutting-edge technologies. Players could gain some insight into the world of robotics, AI, and virtual reality, making them eager to explore and experience more of what Delta has to offer.

Figure 10. A depiction of the adversary 'Robot Jackal'

This experience could be implemented using the following steps:

1. Game Setup & Introduction: Design a dimly lit lobby environment with suspenseful lighting and sound effects to immerse players as they begin their mission in the Delta Robotics lab.
2. Enemy Robot Behavior & Combat Mechanics: Implement corrupted jackals with glowing circuits, health bars, and reactive animations, allowing players to engage them with water guns for takedowns.
3. HUD & Game Indicators: Add a HUD displaying the player's score, number of jackals remaining, water gauge, and CO2 levels to provide essential gameplay information and create strategic challenges.
4. Map Navigation System: Incorporate a map with red markers indicating jackal locations to guide players through the building and enhance exploration.
5. Dynamic Environment: Create corridors and rooms filled with interactive educational elements on robotics, AI, and VR to engage players beyond combat.
6. Progressive Difficulty: Introduce more challenging jackals and environmental obstacles as players advance, increasing the intensity of the gameplay while promoting skill development.
7. Educational Insights & Exploration: Use the game as a tool to familiarize players with Delta University's Robotics lab layout and cutting-edge technologies, blending learning with entertainment.

3.3.8 Direct Teleportation Hotspots (Feature)

An enhancement proposed for the DeltaVR experience could involve integrating a direct teleportation system designed to streamline user navigation and boost informational engagement within the virtual environment. This system may feature visually marked, clickable hotspots positioned throughout the Delta Center. These hotspots could allow users to instantly teleport to key rooms or areas, offering a seamless method of movement that might replace traditional point-to-point walking in VR.

Each significant location within the building could include a unique, easy-to-identify hotspot. When users interact with one, they might be instantly transported to the corresponding space. Upon arrival, an interactive pop-up may appear, providing engaging details about the room's purpose, current functions, and features, potentially through text, images, or short video clips. To further enhance the experience, users could access a floating, interactive map at any time, enabling them to view all available hotspots and jump directly to any room from a central interface.

This teleportation feature might offer multiple benefits. It could greatly improve navigation efficiency, allowing users to bypass less relevant spaces and focus on areas of interest, making the experience more user-friendly, especially for those new to virtual environments. By pairing quick access with informative pop-ups, users may better understand the layout, purpose, and resources of the building, contributing to a more educational and intuitive tour. The ability to explore on demand may also increase user engagement and enjoyment, keeping the experience dynamic and personalized.

For administrators, the system might be designed to allow easy updates to hotspot content, ensuring the virtual environment reflects the most current use and layout of the building. Prospective students, collaborators, or guests could use this feature to gain meaningful insights into the facility, which may influence their interest in visiting, joining, or working with the institution. Highlighting spaces like labs, lecture halls, or collaborative zones may showcase the building's strengths and support broader engagement.

Emotionally, users could feel a sense of control and confidence as they explore. Being able to teleport instantly to points of interest may create excitement and curiosity. The combination of guided navigation and informative content might foster a deeper connection to the space, turning a simple tour into an engaging and memorable learning journey.

This feature can be implemented using the following steps:

1. Clickable Hotspot Placement: Place visually distinct, interactive teleportation hotspots at key points in each significant area of the DeltaVR environment (labs, lecture halls, offices).
2. Teleportation Functionality: Implement a script for hotspots to instantly move users to the corresponding rooms upon clicking, ensuring smooth transitions to enhance navigation.
3. Informational Pop-ups: Create detailed informational pop-ups that trigger upon arrival in a room, featuring text, images, and videos to explain each room's purpose and key activities.
4. Interactive Map Integration: Design a dynamic, accessible map displaying all available hotspots. Allow users to click directly on the map to teleport to any chosen room.
5. User Experience Enhancements: Add a seamless user interface to access the map at any time, giving users control over their navigation and reducing the need for manual exploration.
6. Admin Management & Updates: Develop an admin panel to easily update hotspot placements and associated informational content, ensuring the virtual tour remains current and relevant.
7. Testing & Optimization: Conduct user testing to ensure smooth teleportation, accurate hotspot placement, and engaging information pop-ups. Optimize for low latency and visual consistency to improve user satisfaction.

3.3.9 AI-driven analytics system (Feature)

The AI-driven analytics system in DeltaVR could be designed to integrate with the existing infrastructure, harnessing data from IoT devices distributed across the virtual campus. This system may comprise several layers, beginning with the Data Collection Layer, which could involve numerous sensors and IoT devices gathering real-time data on energy usage, temperature, occupancy, and other key metrics. Following this, the Data Processing Layer might utilize a robust cloud computing infrastructure to handle the raw data, employing machine learning algorithms to analyze and interpret the information. These algorithms could identify patterns and predict future trends, enabling advanced AI to forecast peak usage times, potential equipment failures, and other critical events. The AI might also simulate various scenarios to determine the most efficient strategies for managing energy consumption.

The insights and data processed could then be presented to users through an intuitive and interactive interface within DeltaVR. This interface might include dashboards, charts, and alerts, making it easier for users to understand and respond to the information. The benefits of this system may be substantial. For example, predictive maintenance could help anticipate equipment failures and schedule timely interventions, potentially reducing downtime and extending equipment lifespan. Additionally, the AI might optimize energy usage by adjusting systems like HVAC and lighting based on occupancy patterns and environmental conditions, leading to cost savings and improved comfort. Finally, administrators could gain actionable insights and recommendations, empowering them to make more informed decisions about energy management and resource allocation.

This experience may be implemented using the following steps:

1. Machine Learning Algorithm Integration: Use AI algorithms to analyze collected data, identifying patterns, forecasting peak usage, and simulating energy management scenarios.
2. AI-Driven Forecasting: Enable predictive maintenance by using the AI system to identify potential problems and schedule timely interventions before issues arise.
3. Dynamic Energy Optimization: Automate energy management by adjusting HVAC, lighting, and other systems based on real-time occupancy patterns and environmental conditions for maximum efficiency.

4. Interactive User Dashboards: Design intuitive dashboards and visual interfaces within DeltaVR to display analyzed data, forecasts, and actionable insights through charts, alerts, and reports.
5. Administrator Empowerment: Provide admins with real-time alerts and actionable recommendations, enabling informed decisions on energy management and resource allocation to enhance comfort and reduce costs.

3.3.10 Control-Room (Feature)

The virtual control room in DeltaVR could serve as a centralized hub where administrators may efficiently manage and monitor the campus's energy systems. This immersive 3D environment might replicate a real-world control room, allowing administrators to navigate and interact with various control panels and displays using VR controllers. Within this virtual space, multiple interactive dashboards could provide real-time data on energy consumption, system status, alerts, and historical trends, which may be customized to focus on specific areas of interest.

The control room might also include interfaces for remotely adjusting settings—such as changing thermostat levels, turning off lights in unoccupied areas, or redirecting power as needed. This could be viewed as an advanced continuation of the digital twin functionality described earlier in the application. An integrated alert system may notify administrators of anomalies or issues, like unexpected spikes in energy consumption or equipment malfunctions.

The potential benefits of this setup could be significant: it may enhance monitoring by offering an overview of the campus's energy systems, enable remote management for implementing energy-saving strategies and responding to emergencies, and promote efficiency and cost-effectiveness through data-driven decision-making. Furthermore, the immersive nature of the virtual control room might engage users more effectively than traditional interfaces, simplifying the understanding of complex systems and data.

This experience could be implemented using the following steps:

1. 3D Environment Design: Develop an immersive 3D control room with interactive control panels and customizable dashboards for real-time data visualization.
2. Data Collection and Display: Integrate IoT devices and sensors to feed live data on energy usage, occupancy, and system status into the control room environment.
3. Remote Control Interfaces: Enable administrators to adjust settings (e.g., thermostat, lighting) using VR controllers to manage energy systems directly within DeltaVR.
4. Alert System Integration: Develop an AI-driven alert system that displays notifications for anomalies like equipment malfunctions or unusual energy spikes.
5. Predictive Maintenance: Implement predictive analytics to forecast potential equipment failures or inefficiencies and display recommendations for proactive action.

3.3.11 Asset Management (Feature)

The asset management [9] feature in DeltaVR could allow administrators to remotely monitor and manage the condition and utilization of valuable assets such as VR headsets etc. The system could include a robust tracking mechanism that logs the status, usage patterns, and maintenance schedules of each asset. The design will incorporate a centralized dashboard within the virtual control room, where administrators could view real-time data on asset conditions, receive alerts for required maintenance or potential issues, and optimize space utilization to prevent resource wastage. The dashboard will be interactive, providing detailed reports and analytics that aid in informed decision-making. By leveraging IoT sensors and AI-driven analytics, the system will predict when assets need maintenance, ensuring timely interventions and reducing downtime.

This asset management system can improve asset performance, increase return on investment through better utilization, and enable remote management capabilities. Additionally, it could help in reducing operational costs by preventing unnecessary replacements and repairs, ultimately leading to a more sustainable and efficient management of resources. This ensures that educational institutions can maximize resource efficiency, enhance the learning environment, and achieve long-term sustainability goals.

This experience could be implemented using the following steps:

1. IoT Sensor Integration: Deploy IoT sensors to VR headsets and other assets to monitor real-time usage, condition, and location data for tracking and analysis.
2. Centralized Dashboard Creation: Design and integrate an interactive dashboard in the virtual control room, displaying asset condition, utilization data, and maintenance schedules for administrators.
3. Data Collection and Syncing: Develop a data pipeline to sync asset data in real-time, ensuring that all updates are reflected in the dashboard for accurate and timely insights.
4. AI-driven Predictive Maintenance: Implement machine learning algorithms to analyze usage patterns and predict when assets require maintenance, enabling proactive interventions to reduce downtime.
5. Alert and Notification System: Create an alert mechanism to notify administrators of potential issues, required maintenance, or low-usage assets to prevent costly repairs or replacements.
6. Utilization Optimization: Incorporate analytics that provide insights into space and asset usage, enabling better allocation, rotation, and storage of resources for enhanced efficiency.
7. Reporting and Analytics Tools: Develop detailed reporting and analytics features to generate actionable insights, aiding in informed decision-making and optimizing return on investment.

To enhance the DeltaVR experience, several innovative features are proposed. ProtoFlux offers a powerful scripting tool enabling users to create dynamic experiences, akin to Unity's editor but within the VR environment. Big AI’s Avatar Corridors provide users with a diverse range of avatars from popular shows, enhancing personalization. The Computer Museum integrates detailed 3D models and interactive features to educate users on the evolution of computers. Direct Teleportation Hotspots improve navigation by allowing instant travel to specific rooms, accompanied by informative pop-ups. Virtual Drawing allows 3D drawing and modeling, potentially extending to 3D printing. An AI-driven analytics system and a virtual control room enhance energy management and asset monitoring. District Roboto offers an immersive futuristic cityscape with robots, while Magic Freeze Tag introduces a multiplayer tag game in a maze-like environment, fostering collaboration and excitement.

Some of the experiences present in this chapter are designed to improve user enjoyment, like Visual Scripting, Avatar Corridor, and Computer Museum. Whereas other features like the Teleportation and Control Room simply make the experience easier for users and improve the digital twin experience as well. The main challenge is developing and integrating these experiences and insights into how it's to be done has been included with each experience and feature. Further user testing and feedback can even help in coming up with more ideas on how the DeltaVR project can be improved. ChatGPT was used to generate implementation details for each experience and feature. Each prompt included a description of the experience or feature along with the instruction: "Give steps to implement this experience/feature."

4. Implementation

This chapter contains all the implementation details regarding the features mentioned in subchapters 3.1 & 3.2. Developers who want to learn more about the technical details related to the digital twin features and want to build more such features may take help from this chapter. It is also useful for anyone wanting to add dashboards (showing building metrics) in the application without having strong Unity and VR development knowledge. Subchapter 4.1 describes the difficulty faced in running the existing project's APK on a headset and how the delta building assets were imported into another Unity project to start development of the digital twin features. Subchapters 4.1.1 and 4.1.2 explain how the project was imported into Unreal as well as in Godot Engine to allow the project to be developed in these engines in the future. Subchapters 4.2 and 4.3 give details about how the live weather data from OpenWeatherMap, the building data (CO_2, energy levels & temperature), and classroom schedules have been integrated, respectively.

4.1 Exporting Delta Building Assets

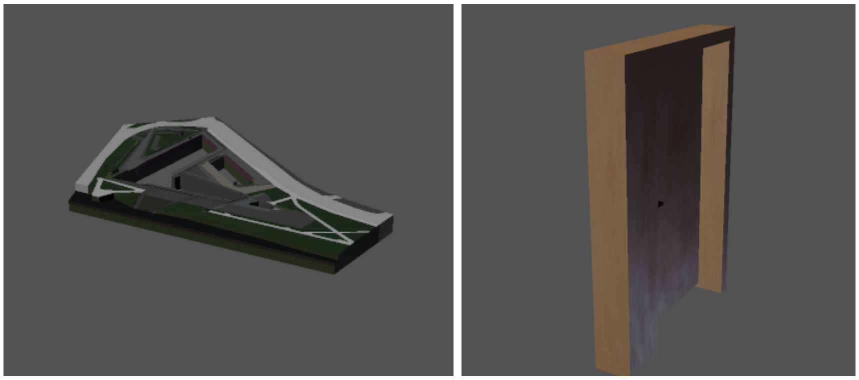


Figure 11. Building and Objects as Prefabs

4.1.1 Exporting to another unity project

The latest APK present for the DeltaVR project that was from the DeltaVR Multiplayer: 2.0 installation completed in 2023 and a problem arose in trying to run that apk in the headset. The apk would launch, show the main menu however it would not go further to the virtual experience itself. After spending a lot of time looking into the issue and not being able to resolve it, we decided to shift the existing DeltaVR project to another unity project to start development.

To shift the project, first it was necessary to understand the structure of the project. The entire building is segmented into various parts, such as the first floor, second floor, and so forth. Each of these larger segments has its own models, while smaller objects like posters in labs and doors are also included. All components of the building are either as prefabs or are in the fbx[10] file format, which allowed them to be seamlessly integrated into another project by just importing the files. To maintain the building's precise appearance, a wide array of materials is utilized, which I also exported alongside the models.

10 https://www.autodesk.com/products/fbx/overview?msockid=323db780e6ea6b433a11a2d5e7bd6a58

4.1.2 Unreal Engine

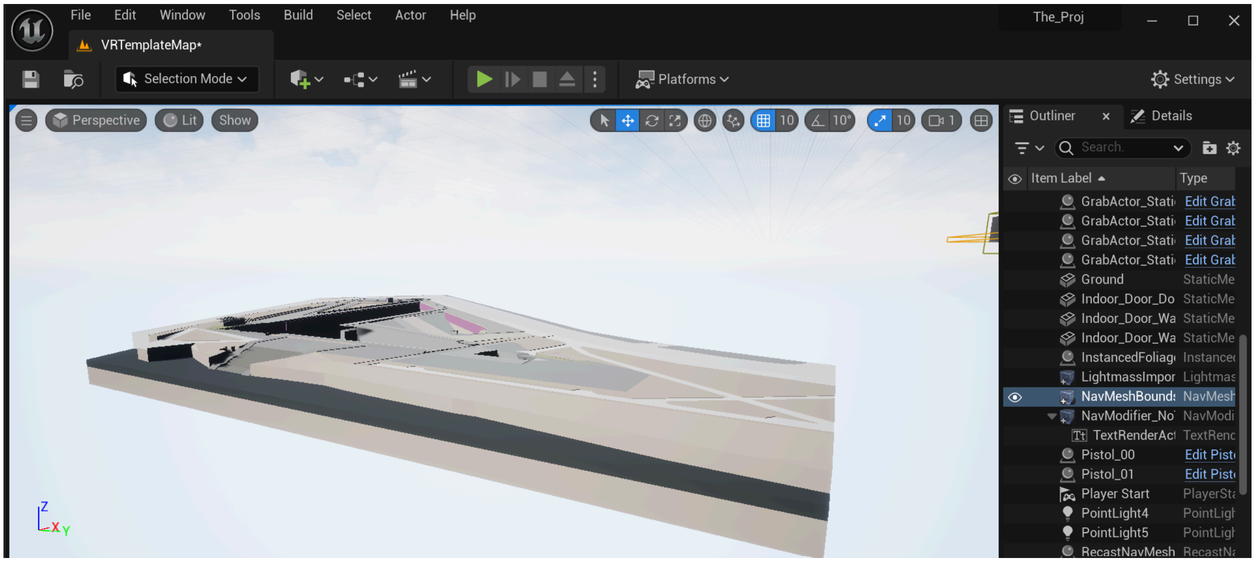


Figure 12. A scene in Unreal Engine showing the Delta Building's ground during the export process

This chapter explains how I shifted the DeltaVR project from Unity to Unreal Engine through clear-cut steps to guide future developers interested in shifting the project to Unreal Engine. I shifted the project using the following steps:

1. Navigate to the Content Browser: In the Unreal Engine Editor, open the Content Browser, where you manage your assets.
2. Import the FBX File: Click the "Import" button in the Content Browser. Select your FBX file from your computer. The FBX Import Options dialog will appear.
3. Set Import Options: Configure the import settings according to your needs:
    a. Skeleton: If you are importing a skeletal mesh, choose the skeleton to use or create a new one.
    b. Animations: Import animations if your FBX file contains them. Materials and Textures: Ensure the materials and textures are set up correctly.
    c. Geometry: Adjust settings related to the geometry, such as scale and orientation.
    d. Collision: Set up collision options if needed.

4. Complete the Import: Click "Import" to finalize the process. The FBX file and its associated assets will be imported into the Content Browser.
5. Use the Imported Assets: You can now drag and drop the imported model into your scene, apply materials, and set up animations as needed.

Steps to convert a Unity prefab to fbx file format:

1. Go to the package manager in the Unity editor.
2. Select ‘Packages: Unity Registry’ from the top left.
3. Search and install the ‘FBX converter’ package.
4. Right-click on the prefab and select ‘Convert to FBX prefab variant’
   a. Optional: Configure conversion options
5. The prefab will be converted to an .fbx file in your desired location.

4.1.3 Godot

The delta building model is in the .fbx file format in the Unity project, and this chapter contains the steps to import an fbx file/model to Godot.[11]

1. Download the plugin[12] according to your operating system, extract the content, and copy the path of the directory where the license is.
2. Click on the editor from the top options and select ‘Configure FBX Importer’.

[11] https://godotengine.org/
[12] FBX Import (godotengine.org)

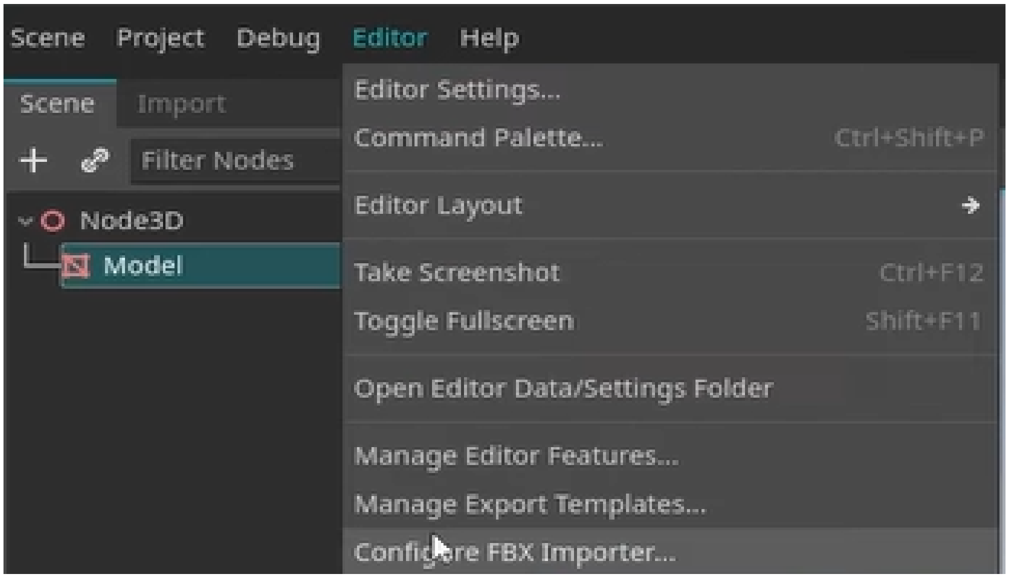


Figure 13. Godot layout indicating showing configure FBX importer option

3. Click on the browse option.

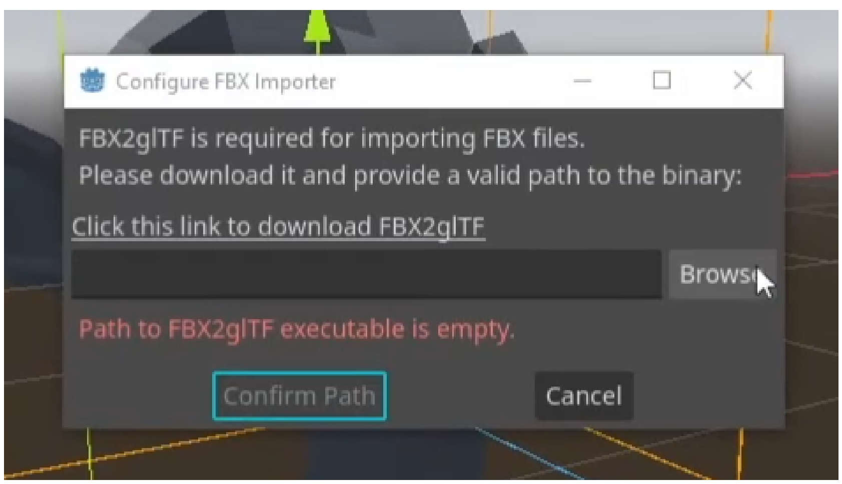


Figure 14. Option to browse for license in Godot

4. Add the path to the license, click on the added license and press open.

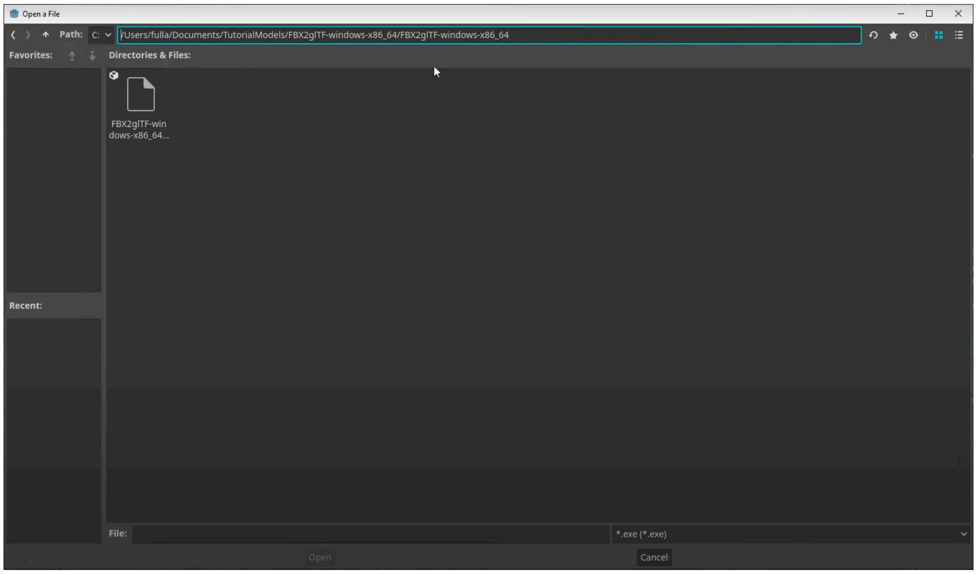


Figure 15. Option to select the license in Godot

5. Drag your .fbx file and use it in the scene.

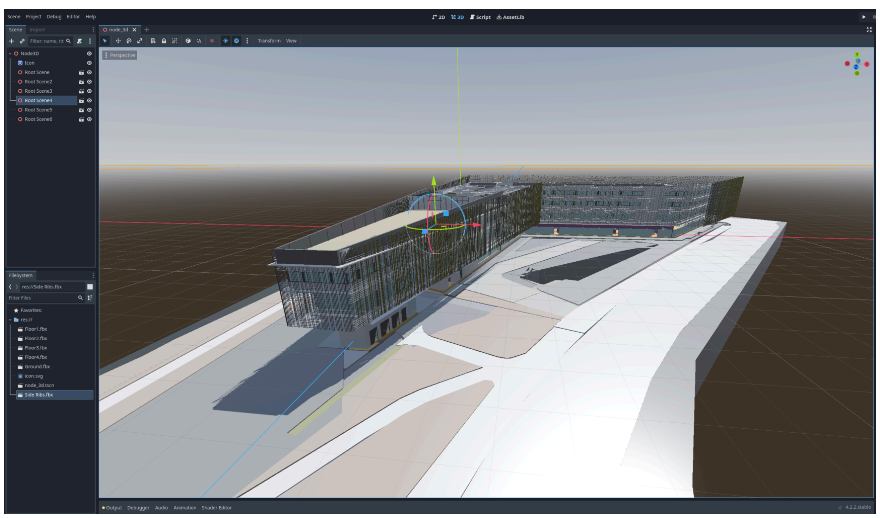


Figure 15. Delta Building FBX imported in Godot

Since the building model is in .fbx file format, which is understandable by various 3D development engines, it’s fairly easy to shift the model from Unity to another engine. The complexity in the development process depends on the engine and the features to be developed. It’s best to shift the project to Unreal Engine for development in the future, as it can provide much better graphics, improving the overall user experience.

4.2 Weather API Integration

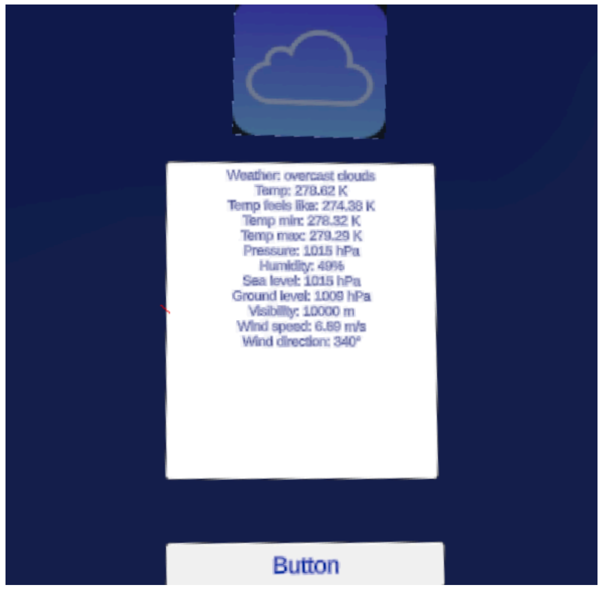


Figure 16. A dashboard in the DeltaVR application showing the live weather data

The chapter contains details about how I integrated live weather data with the virtual building. The steps to achieve this included obtaining the weather data, scripting the data fetch and display, and setting up interactive UI elements within the VR environment, which are detailed below.

1. **Obtain Weather Data from OpenWeatherAPI**
    a. Create an Account: First, I created an account on the OpenWeatherAPI platform.
    b. Get API Key: Upon registration, I obtained an API key, which is essential for making authenticated requests to the OpenWeather API.
    c. I Used the API Key to get the URL to make the API request for getting the weather
2. **Script to Fetch and Display Weather Data**

a. A script was developed that fetches weather data from the OpenWeatherAPI and displays it in an Input Field (TMP_InputField[13]). A button click triggers a coroutine that sends a web request, retrieves the data, and updates the UI with the result.

3. **Adding UI Elements in Unity**
   a. To display the weather data, I added several UI components to the Unity scene:
   b. Canvas: I added a Canvas to the scene to serve as the container for the UI elements.
   c. Button: I added a button to trigger the weather data fetch.
   d. Text Display: I added a TextMeshProUGUI[14] element to display the fetched weather data.
   e. Attach Script: I attached the weather data fetching script to the canvas in the scene.
   f. To make the button interactable in a VR environment, I used Unity's XR Interaction Toolkit[15].

[13] https://docs.unity3d.com/Packages/com.unity.textmeshpro@1.3/api/TMPro.TMP_InputField.html
[14] https://docs.unity3d.com/Packages/com.unity.textmeshpro@1.1/api/TMPro.TextMeshProUGUI.html
[15] https://docs.unity3d.com/Packages/com.unity.xr.interaction.toolkit@3.0/manual/index.html

4.3 Building Data Integration

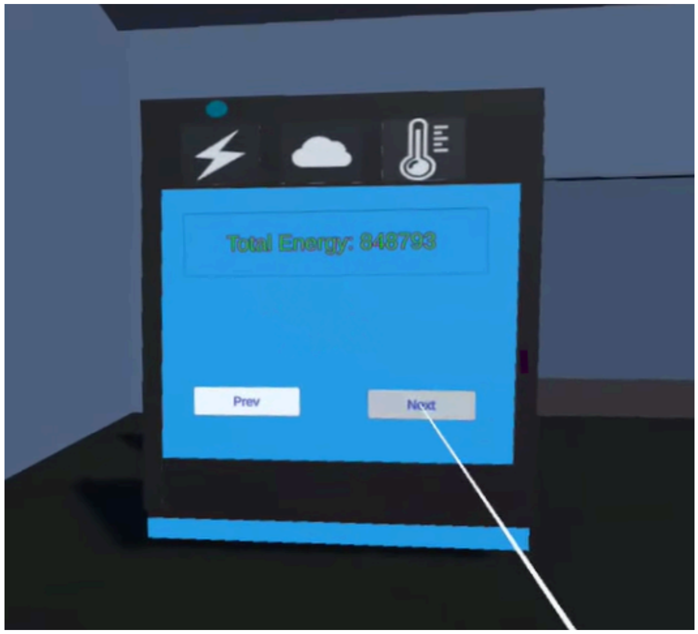


Figure 17. A Dashboard showing building data

The chapter describes how by integrating data retrieved from the Delta building's influxDB, the DeltaVR system provides a user-friendly interface for monitoring building automation metrics such as temperature, CO2 levels, and energy consumption. Chapter 4.3.1 explains the implementation details of integrating the building data with DeltaVR using Unity. Chapters 4.3.2, 4.3.3, and 4.3.4 explain how to add a new dashboard to the building using the global configuration file and the data model. Chapter 4.3.5 explains how the global configuration can be updated at runtime to get the desired data. A VPN connection is required to retrieve the data from the influxDB and the steps to establish a connection on the quest are present in the appendix II.

4.3.1 Implementation Process

An example of the URL that is used to retrieve the data:
*http://**username:password**@172.17.67.20:8086/query?db=delta&q=SELECT%20*%20FROM%20*

%22KogEN%22%20WHERE%20time%20%3E%3D%20%272024-01-01%27%20AND%20time%20%3C%3D%20%272024-01-02%27 (Username and password are to be replaced with the actual credentials).

This command sends a GET request to the InfluxDB server, authenticates using the provided credentials, and queries the database delta for entries in the KogEN measurement between January 1, 2024, and January 2, 2024. The query results are returned in JSON[16] format, suitable for further processing.

Displaying in Delta VR

The Delta VR app utilizes the Unity framework to handle data fetching, asynchronous processing, and user interface management, creating a dynamic and user-friendly data display feature. Below is a detailed breakdown of the implementation:

1. Fetching Data Using URLs with Credentials: Data retrieval is performed using URLs specified in a JSON configuration file. These URLs include credentials and point to specific datasets such as energy consumption, CO2 levels, and temperature. The URLs query a database, like InfluxDB, with the necessary credentials embedded for authentication.
2. Extracting and Displaying Numerical Data: Once data is retrieved, numerical values are extracted from the raw data using Regular Expressions (regex). This allows for the precise extraction of numerical data along with their respective units (e.g., °C, °F). The extracted data is then parsed and displayed on a user interface created within Unity, utilizing text fields to present the information.
3. Maintaining Up-to-Date Information: The VR app fetches the most recent energy data using SQL[17] queries specified in the configuration URLs. To ensure the data remains up-to-date, the app uses Unity's coroutine feature to fetch the data periodically, every two minutes. This is implemented using Unity's StartCoroutine method, allowing the data-fetching process to run asynchronously without freezing the main application.

---

[16] https://www.w3schools.com/whatis/whatis_json.asp
[17] https://www.w3schools.com/sql/sql_intro.asp

4. User Interface Design: The user interface features a single display field named 'TheOutput' where the data is shown. To navigate through different data sets (energy, CO2, temperature), two buttons labeled 'Next' and 'Prev' are implemented, allowing users to cycle through the displayed data interactively.
5. Configuration File for Easy Adjustments: All URLs, polling rates, and information lines are stored in a global configuration file. (Assets/GlobalConfig.json) This centralizes all the settings, making it easier to adjust parameters without modifying the main script.
6. Room and Host Specific Data: The data retrieval process involves querying for room and host-specific data. The configuration file includes URLs tailored for specific rooms and hosts, ensuring that the information displayed is contextually accurate for the environment being simulated.
7. Displaying Data on Canvas: The extracted and processed data is displayed in the VR environment on a decorative text display on a canvas with a matching background color. The canvas serves as an overlay in the VR scene, ensuring that data is easily readable and aesthetically integrated.
8. Fetching and Displaying Icons: Icons representing different weather conditions or building data states (such as sun, rain, clouds for weather; various energy states for building data) are fetched and displayed alongside the numerical data. These icons enhance the visual representation of data, making it more intuitive for users.
9. Navigation Buttons for Data Display: To navigate through different datasets, such as different rooms or types of data (energy, CO2, temperature), the VR app includes 'Next' and 'Previous' buttons. These buttons allow users to cycle through the available data displays, providing a view of the building's status.
10. Adding New Dashboards: The configuration for data retrieval and display is managed through the JSON file. This file includes parameters such as polling rates, information lines, and URL sets for different data types and locations. To add new dashboards, the JSON configuration file is updated with new URL sets and data types. The VR app reads this configuration file during initialization and dynamically adapts to the new settings.
11. Data from JSON: Building and Weather: Data integration involves retrieving information from JSON for both building and weather data. This includes parsing the JSON configuration file to extract URLs and credentials, as well as parsing JSON responses

from web requests to retrieve specific data points like temperature, humidity, and energy consumption.

12. Creating Classes and Template Readers: To manage the data effectively, classes are created to represent different data types and configurations. For example, classes like Config, PollingRates, UrlSet, and OpenWeather are defined to match the structure of the JSON data. A template reader method is implemented to parse the configuration file and instantiate these classes with the appropriate data, ensuring a structured and maintainable approach to data management.

4.3.2 Add Dashboards in Different Rooms

The project contains a global configuration file that contains the URLs used to retrieve the desired building data. It contains separate URLs to retrieve different values for energy, CO2, and temperature levels of different rooms in the Delta Building. This can be updated before building the executable of the project. The picture below shows the global configuration file (Global-Config.json) present in the assets/StreamingAssets folder and which can be edited to add URLs that can be used to make a new dashboard. It shows an array of URLs and their corresponding value names. To add a new dashboard, first add a new set of URLs corresponding to Energy, CO2 & Temperature, to the “urlSets” array. Then change the ID (using the data model in 4.3.2) of the host device (highlighted with the red squares) and note the index number of the set. e.g, in this case, the index number would be 1.

Figure 18. The global configuration file

Add a new Dashboard in the application by following these steps:

1. Open the DeltaVR project in Unity.
2. Open the DeltaVR Scene present in assets/scenes
3. Duplicate the 'Test' object in the scene by right-clicking

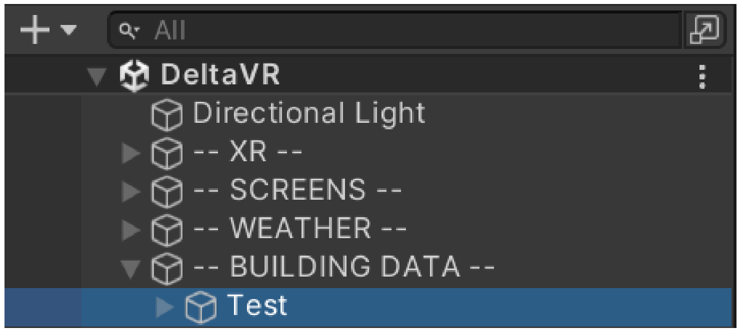


Figure 19. DeltaVR project hierarchy in unity

4. Move it to the desired room. (Optional)
5. Duplicate the GetMethodTest script in **assets/scripts**, add a desired name, and inherit the GetMethodBD1 class.
6. Add the script to the 'canvas text'.
7. Change the names of the Output canvas and the next and previous buttons.

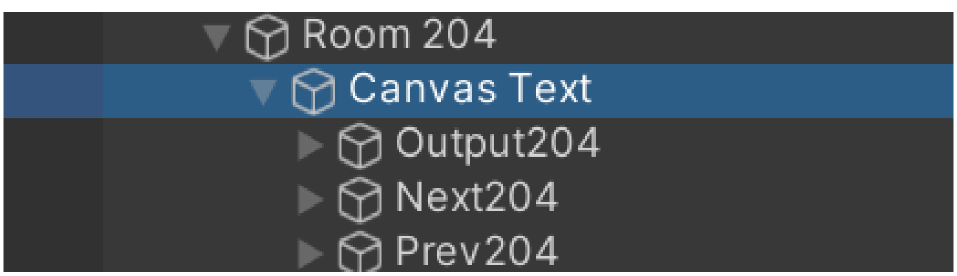


Figure 20. DeltaVR project hierarchy in unity

8. Update the following variables. 'urlSetIndex' is the transcript number of the set containing data to be displayed, 'outputName' is the name of the output canvas, 'nextName' and 'prevName' are the names of the next and previous buttons according to

the previous step.

```
void Start()
{
    urlSetIndex = 1;
    outputName = "TheOuput1";
    nextName = "Next1";
    prevName = "Prev1";
    base.Start();
}
```

Figure 21. Piece of code for displaying data in dashboards

9. Get the host ID of the data to be displayed (from the data model in 4.3.2) and add it in the global configuration (As mentioned above)
10. Build the project and get the APK.
    a. File -> Project settings -> Select Android platform -> Build

4.3.3 Retrieving Building Data From Data Model

This section outlines details about how the university's building data is structured and how it can be retrieved from the InfluxDB. It explains how the data model[18] can be used in converting DeltaVR into a digital twin. It covers shorthand measurement names for energy, CO_2, and temperature, collected from specific host devices in various rooms within the Delta building.

The data model used in the building automation system consists of various shorthand Estonian measurement names representing different types of data:

1. KogEN: Total energy consumption.
2. DP, Dp1, DP2: CO2 levels.

18 https://docs.google.com/document/d/1XPmH8e2SZ4wlxw9SChF8GcscefAsuSyoL5iuCPSKeUk/edit?usp=sharing

3. TSu, TSu1, TSu2: Temperature readings.

Each data point in the model includes a "host" field, specifying the device from which the data originates. This is essential for mapping and identifying the source of measurements within the building automation system.

This URL example shows how Measurement Name (DP2) and Host ID (110530530**)** can be used to obtain the desired Energy, CO2, or temperature values:

http://username:password@172.17.67.20:8086/query?db=delta&q=SELECT*FROM"<Measurement_Name>" WHERE "host" = %27<Host_ID>%27 ORDER BY time DESC LIMIT 1

(DP2 & 110530530 combination means it is the CO2 value time series for the 2020 Robotics room)

4.3.4 Updating the Global Configuration at Run-time

This chapter explains the implementation details of how the global configuration can potentially be updated at run time by the user through a UI in the DeltaVR application. To allow users to update global configuration values, you'll first need to create a user-friendly interface using Unity's Canvas system. The Canvas system provides the foundation for building interactive UIs in Unity, allowing for easy integration of various UI elements like buttons, text fields, and labels. In this case, your Canvas will include multiple input fields where users can enter the values that will update the global configuration (such as URLs, API keys, or other data). Labels are to be added next to these input fields so users can easily understand what data they are modifying (e.g., "Room 2049 Weather URL"). Additionally, a Button labeled "Save Configuration" should be provided to trigger the saving of data. Make sure the UI elements are laid out neatly and are easy to navigate. You can customize the input fields to accept different types of data, such as strings or integers, depending on the specific values you need to update in the global configuration.

Once the UI has been set up and users can input new values, the next step is to capture this data using a Unity C# script. This script will interact with the UI components, particularly the InputFields. When a user inputs new values and presses the "Save Configuration" button, the script will retrieve these values via the InputField's text property (for strings) or convert them into the required data type (e.g., integers). You can achieve this by assigning the InputField and Button references in the Unity Inspector or by accessing them programmatically. The script will also include methods that are triggered when the button is clicked, collecting the data from all the input fields and preparing it for writing to the global configuration. The goal here is to ensure the script correctly receives all the data from the UI and validates it, ensuring it is in the correct format (e.g., valid URL or API key) before proceeding to update the configuration file.

The most significant technical challenge lies in writing the updated values into the global configuration file, typically stored as a JSON file. Unity's JsonUtility class will help serialize and deserialize the data to and from JSON format. The process involves loading the existing configuration file, updating it with the new values provided through the UI, and then saving the modified JSON back to the file system. You'll begin by reading the current global configuration file using File.ReadAllText(), and then use JsonUtility.FromJson() to convert the JSON string into a C# object. Once you have this object, you can update its fields with the new values from the InputFields.

After making these changes, you serialize the object back into JSON format using JsonUtility.ToJson() and overwrite the original configuration file using File.WriteAllText(). The challenge is ensuring that the file is accessed correctly, handled safely (i.e., proper error handling for file access issues), and that the data written is valid, especially when the application is running on different platforms. For larger or more complex JSON files, you might need to consider using external libraries like Newtonsoft.Json for more advanced JSON handling capabilities.

4.4 Schedule Integration

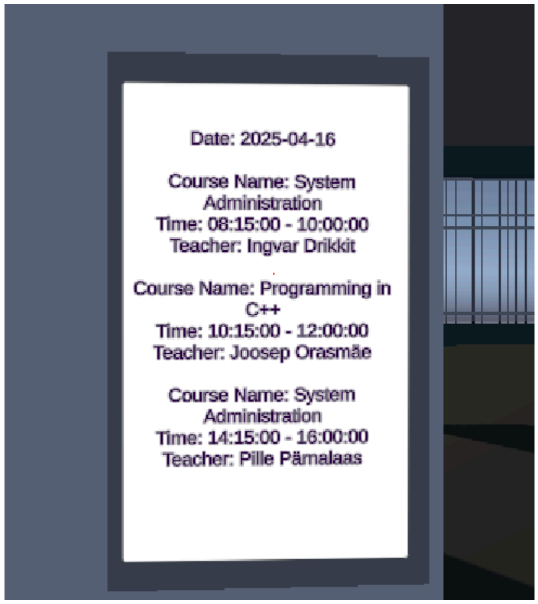


Figure 22. A board showing the schedule information of a classroom

This chapter contains the details about how I integrated the schedule data of different classrooms into the virtual model. The schedule data for rooms in the Delta building is retrieved using the ÕIS 2 API. This API allows for querying the timetable of specific rooms by sending a POST request with details like the building code, room number, and date. In Delta VR, this data is fetched by making an API call to the ÕIS 2 endpoint, which returns the room's schedule in JSON format.

In Unity, the API call is implemented using a coroutine that sends a POST request to the ÕIS 2 API. The request includes parameters such as the building code (e.g., "NAR18OH" for the Delta building), the specific room number, and the desired date. Once the API response is received, the schedule data is retrieved as a JSON string. This data can then be parsed and used within the VR environment. The schedule data is displayed on interactive panels within the Delta VR environment. Each panel corresponds to a specific room and shows the schedule for that room.

Overall, the whole chapter 4 helps to make future development on the whole project, specifically the digital twin part, easier and helps in making more use of the digital twin features without having technical knowledge of Unity, through clear-cut steps. The process involves understanding and retrieving building data from the university's InfluxDB, fetching it using a script in Unity, and displaying it using Unity's UI components. The chapter also helps to understand how the building data is stored and retrieved in detail, which can help in understanding and integrating more complex data in the future. There's also insight into how the process of utilizing the digital twin features can be made easier through a UI in the application.

5. Testing & Presentation

This chapter is dedicated to the testing of the application and will help anyone to test or present this application to an audience. Subchapter 5.1 gives details about the initial usability testing, and it contains subchapters that explain the methodology used to invite the participants, the design of the questionnaire, and the results of the testing. Subchapter 5.2 contains details of the second presentation session and how the application can be presented to an audience to receive their feedback. The process of identifying the audience, inviting them, making tailored questions for different types of testers to get feedback, and methods to record the feedback are presented in the chapter, along with the results of the second presentation session.

5.1 Initial Usability Testing

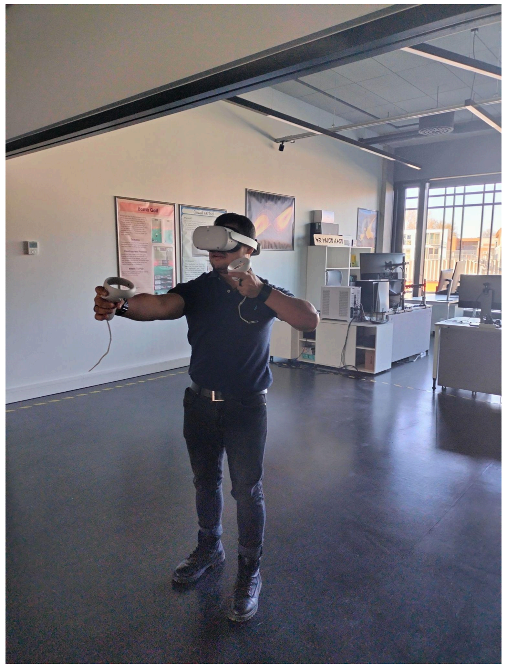


Figure 23. A student at the initial usability testing

This chapter presents the findings of the initial qualitative usability testing conducted on 5th March, 2024, for the Delta VR app in the Computer Graphics and Virtual Reality lab[19] (Room 2007) from 11:00 am to 6:00 pm. Usability testing refers to testing the application with actual users to see how easy it is to use and to find any issues that might arise. Doing this helps make the application easier and better to use. The purpose of this testing was to identify areas for improvement and potential additions to enhance the user experience, particularly focusing on eliciting desired emotions from the target audience [7].

5.1.1 Methodology and Feedback Questionnaire

Conducting usability testing in a controlled environment is recommended for obtaining more detailed and reliable feedback to enhance the user experience of an application, and the use of a

[19] https://cgvr.cs.ut.ee/

questionnaire is essential [6]. Feedback obtained from expos may not be as accurate due to time constraints, lack of privacy, and reluctance among participants to offer critical feedback in a public setting. Six participants were recruited, and it involved them trying out various features of the Delta VR app and providing their feedback through structured questions and open-ended discussions. The participants were asked about their comfort level, opinions on the application, experience with specific features like the archery range, and their opinions on adding new experiences.

The feedback questionnaire comprised several subjective inquiries aimed at gathering user perspectives on various aspects of the application. Participants were encouraged to share their suggestions and overall opinions regarding the application. Additionally, participants were asked if they had experienced any feelings of sickness or nausea during their interaction with the application. Since the archery range is one of the most prominent experiences in the application, feedback was sought regarding its potential improvement. Furthermore, opinions were solicited regarding the addition of exploratory experiences within the application.

Exploratory experiences were explained as activities designed to help users learn more about the different areas in the Delta Centre, such as themed games centered around the labs. Participants were asked whether they believed such experiences would enhance the application's overall experience. Additionally, the questionnaire delved into users' emotional responses, aiming to discern the range of emotions elicited during their interaction with the application. Users were asked to express their emotional experience in a single word or phrase, allowing me to analyze the effectiveness of features designed to evoke specific emotions.

5.1.2 Results

During the VR session, participants reported varied experiences with comfort. One participant mentioned having issues due to their glasses, while another reported discomfort due to eye problems. Suggestions were made for accommodating colorblind users. In general, participants expressed satisfaction with the movement method, and they appreciated the futuristic space background for its ambiance. Suggestions were provided to enhance engagement, such as incorporating collection quests and timed challenges. Feedback on the archery range experience was positive overall, with participants finding it exciting. Participants responded positively to the idea of adding a lab exploratory experience to learn more about the labs and inhabitants of the Delta Centre.

Participants primarily responded with words such as "exciting," "amazing," and "interesting" when prompted to describe their emotional experience in one word. Both the verbal responses and participants' behaviors were analyzed to gain insight into the emotions elicited during their interaction with the application. When asked about their desired emotional responses during the VR experience, participants reported a range of emotions, including excitement, interest, enjoyment, confidence, creativity, and focus. These emotional responses align to create an engaging and immersive experience for users. The initial qualitative usability testing of the Delta VR app provided valuable insights into user preferences, comfort levels, suggestions for improvement, and the results were recorded in the table in Figure 3. The purpose of the findings was to help guide further development efforts to enhance user experience and achieve the emotional goals set for the application.

Table 1. Do, Be, Feel Table

Who (Role)	Do (Function)	Be (Quality)	Feel (Emotion)
Students	Arrow always reach target smoothly	Scalable	Excitement, Challenge, Accomplishment, Satisfaction
VR enthusiasts	Spray looks smooth when hand is moving too fast	Interactive	Creativity, Freedom, Expression, Realism
Teachers	Players Collide with all doors	Educational	Realism, Immersion
	Players Can open all doors	Compatibility	Realism, Immersion
	Player don't fall off the map	Navigable	Realism, Immersion
	Fix Teleport to up floors by aiming at ceiling	Accessible	Realism, Immersion
	Add time and map to the collectible game	Adaptable	Adventure
	Robot Shooting Game		Thrill, Adrenaline Rush, Action, Strategy
	Change Space background		Realism, Immersion
	Fix multiplayer delay issue in archery range		Excitement, Challenge, Accomplishment, Satisfaction

5.2 Presentation Guide & Results

The application can be presented to an audience through the steps in the following subchapters.

5.2.1 How to Install APK on Quest

1. Download the latest APK from https://drive.google.com/file/d/1tyun4oiBP42rBlKIjCduumcCLx_qefrR/view?usp=sharin

g

2. Install Meta Quest Developer Hub (MQDH), launch it, and make an account/login.
3. Connect the Quest to the PC using a USB cable & accept the prompt to allow USB debugging in the headset.
4. Select Device Manager on MQDH, find your device, and select ‘Add Build’ to browse and select the APK file on your PC, or drag the APK file into MQDH to install it in the headset.
5. In your headset, click on ‘Apps’, then on the top right drop-down, choose ‘Unknown Sources’ to find your APK.

5.2.2 Identifying the target audience

The presentation can be carried out at once for an audience or separately for individuals. For carrying it out for individuals, look at section 5.1 for details. In the latter case, when identifying the target audience for your application, first consider its purpose and who would find it useful. This app is designed for virtual reality and is based on digital twin technology so potential users could include individuals who enjoy VR experiences or VR Developers who need it for learning, entertainment, or training. IOT developers and enthusiasts would also be potential audiences. Think about the general age range of your audience, whether young students, adults, or professionals. Also, determine whether the app is aimed at beginners, intermediate users, or experts. Tech enthusiasts who are curious about new technologies like VR, gamers who enjoy interactive experiences, new potential students of the University of Tartu, and educators or trainers who seek to enhance their teaching with technology could also be part of the target audience. If in any case, it aids learning, students in schools, colleges, or universities could benefit, especially those in relevant courses or departments.

5.2.3 Invitation and Booking a Room

To present individually, any room in which a person can be hosted can be used. To present to a group of audience at once, a larger room is to be used (approx. $12m^2$). To promote the DeltaVR presentation effectively, several strategies can be utilized. Generally, events can be advertised through social media platforms such as Facebook, Instagram, and X, as well as through messaging applications like WhatsApp or Telegram. Engaging with online communities and forums focused on Virtual Reality, game development, or related technologies can also help reach a wider audience. For an educational audience, it can be beneficial to connect with academic departments in schools or universities. Teachers and professors can share the event details with their students, ensuring the invitation reaches individuals with relevant interests and expertise.

For this presentation, specific target audiences can be engaged using tailored approaches. Potential students interested in the University of Tartu can be invited through the university's social media groups, such as those on Facebook. VR and game development enthusiasts can be reached through platforms like Discord, particularly in communities such as EstonianXR. Students of the Distributed Systems Department, especially those enrolled in courses related to Digital Twins and IoT, are another key audience and can be informed directly through course announcements. Additionally, personal messages to family and acquaintances can be used to extend the invitation to individuals who may find the event interesting or beneficial.

To invite the target audience, a feedback form can be used, such as a Google Form, to allow participants to sign up for the presentation and share their feedback afterward. The form can gather basic details like their name, age, interest in VR, and preferred presentation time. An example is present in the appendix III.

.

5.2.4 Feedback to be taken

To figure out how to take feedback, start by identifying the current and newly implemented features that might benefit from user opinions. For example, the app involves a "digital twin" concept (a virtual model of something in real life), so that users can be asked how well that feature works and what could make it better. Similarly, look at the user interface and ask if it's easy to navigate or if there are areas that could be made more user-friendly. Some questions could be asked of every user regardless of their technical background, others would be more targeted, e.g, for individuals with a VR, IOT, or Software Development background.

Once you've identified the areas for feedback, you can create specific questions to ask users. For example, from every individual you could ask questions similar to:

1. "How easy was it to use the app?"
2. "What additional features would make the app more useful to you?"
3. "Is there anything about the app's design or layout that you found confusing?"

For VR, IOT, and Software developers, you can ask questions similar to:

1. "How can the virtual realism of the app be improved?"
2. "What improvements would make the digital twin feature more realistic or engaging?"
3. "Was there anything missing from the app that you expected to find?"

You can also gather feedback on how the app performs, whether there were any technical issues, and if users felt satisfied with the overall experience. Using these specific questions, you can better understand what users like, what they don't, and what could be changed to improve their experience. You can collect this feedback through surveys (like Google Forms), one-on-one interviews, or informal discussions, depending on what works best for your audience. Compare the feedback from different experience levels to see how familiarity with VR affects opinions and usability.

5.2.5 Presenting

The first step is to have clarity about the new features that are to be presented. For the participants who don't have experience in VR, e.g., the potential students and friends, it's best to give a demo of how to move about and operate in the virtual environment. This can be done by casting the headset's view to your laptop and giving an overview of how to proceed, so they have an idea of how to use the controllers and navigate in the app.

Steps to achieve casting:

1. From your Meta Quest headset, press the Oculus icon on your controller to open the

universal menu.
2. Select Camera, then select Cast.
3. Visit oculus.com/casting, log in, and start.

For the participants who already have experience in VR, a verbal explanation can suffice. All the participants' experiences should be monitored through the laptop screen to guide them if they face any obstacles.

5.2.6 Building Personas and Recording Feedback

The testers are to be asked about their fields and study levels, and then can be requested to take a picture as well. Then they put on the headset to try the application and can answer the questions later on. The feedback can be recorded on a Google document under separate headings for each tester. It is to be made clear what the questions are and what the feedback related to them is. When building personas and recording feedback from testers, you want to gather information about their educational and technical background and how they interact with your app. Here's how you can do it generally, for all testers:

1. Ask Testers About Themselves: Start by asking testers about their background, such as their field of study, job, or area of interest, and their current study level (e.g., high school, university, professional). This will help you understand what kind of people are using your app and how it fits into their lives or work.
2. Optional Picture: If they are comfortable, you can request that they take a picture. This helps to give a visual understanding of the persona you're building, but it's not mandatory if testers prefer not to.
3. Test the App: Have the testers put on the VR headset and try the application. Let them explore and interact with its features. To a certain extent, it's important to let them use it without too much guidance so you can see how easy to use the app is for new users.
4. Ask for Feedback: Once they've finished testing, ask them questions about their experience. You can ask everything mentioned in the subchapter 5.1.5 'Feedback to be taken'.

5. Record the Feedback: Create a Google document where you record the feedback. Make sure to have a separate section for each tester. List their answers under each question so it's easy to see which feedback relates to which question. This makes it organized and easy to refer back to later.

By doing this, you'll build a clear picture of who your users are and get valuable insights into how to improve the app from their perspective.

5.2.7 Results and Areas for Improvement

One of the presentations was held on 10th September, 2024. For conducting it, the CGVR Lab was booked for 4 hours. During these presentation sessions, a total of 4 participants showed up. An example of the feedback form can be found in Appendix III.

In this presentation session, I recorded the following feedback through the questions. Along with the feedback, I mentioned potential problems associated with implementing them and their possible solutions.

There must be signs and boards to help the user know what to do and where to go. Signs may lead to find dashboards as users, especially those unfamiliar with the Delta building, may feel lost or disoriented when navigating the virtual environment. There should also be a mention of the place where the users are right now. To address this, interactive signs and wayfinding tools such as digital arrows or mini-maps can help guide users to important areas like dashboards. Additionally, "You are here" markers or a 3D map could assist users in understanding their current location and can be implemented by dynamically updating the user's position on a 2D or 3D minimap using the player's coordinates in the scene. To handle the technical aspects, a pathfinding system in Unity or similar VR tools can be used to automatically direct users via visual aids or cues. For pathfinding specifically, Unity's NavMesh[20] system or waypoints can be utilized to guide users, with visual aids like arrows or lines projected onto the environment, showing the route to their destination in real-time.

[20] https://docs.unity3d.com/2020.1/Documentation/Manual/nav-BuildingNavMesh.html

Currently, the next and previous buttons are included; it would be better if there were different buttons for different energy sources. e.g, a button named "Temperature," "Humidity," or "Energy Usage," allowing for quick access without cycling through unrelated data. This can be achieved through using Unity's UI Canvas system. A big screen can be added to display the whole data with charts and graphs that will be interactive and help the user visualize all the relevant data in one place, instead of different rooms. Use a large interactive screen within the virtual environment where users can access all data in one place, similar to a real-world command center. Unity's UI Canvas[21] system can be used in this case as well, to render this large screen, and touch or gaze controls can make interacting with graphs and data visualizations easy for VR users.

A shooting game idea was given, similar to bow hunting. Instead, the user suggested having flying birds outside the building, which could be hunted. Implementing a shooting game involving birds around the Delta building poses technical challenges, such as creating complex AI and ensuring seamless integration with the platform's core building management features. Solutions include using pre-built AI or procedural flight algorithms to manage performance and placing the activity in designated areas like rooftops to avoid interference with core functions. Another idea suggested was to give teleport options for different places. This idea is also similar to the one in subchapter 3.3.4. Teleportation between areas could make navigation faster, but seamless transitions are crucial for maintaining immersion. Unity's built-in teleport system or a third-party VR toolkit like VRTK, combined with smooth fades or scene transitions, could solve this issue while allowing users to explore freely.

Another suggestion was to add a Labyrinth/Maze Experience inside the Delta building to give a sense of adventure and mystery. Designing an immersive, yet functional maze within the structure of a real-world building may clash with the intended purpose of the digital twin. It could be added as an optional, gamified experience with simple puzzles related to building management, offering rewards within DeltaVR. Yet another idea was suggested by the same participant, which is to Build Your robot, and is similar to the experience and mechanics building mentioned in subchapters 3.3.1 and 3.3.10. This mechanic, while fun, would require careful

[21] https://docs.unity3d.com/2022.1/Documentation/Manual/class-Canvas.html

integration into the platform. A modular system can be implemented by creating different robot parts as separate 3D models or prefabs, such as arms, legs, or sensors. Each part would have specific attachment points, known as "snap points" or "anchor points," where users can connect it to other parts. This could then be connected to building operations or environmental controls. A quiz experience related to computer science was also suggested. The experience could enhance educational value, but it should be integrated into learning modules focused on building operations or digital twin technology to remain relevant to DeltaVR's goals. Unity's UI system could be used to create interactive and educational quizzes that align with the platform's objectives.

The testing and presentation helped in learning how to get feedback from different potential users on new features. Details of the testing carried out by Tamm [1] in the previous installation of the project helped me to carry out the tests. The usability testing helped me to gather insight into which current experiences in the application are eliciting the desired emotions, such as joy and thrill. This also helped in identifying potential experiences that can be added. (Mentioned in Subchapter 3.3)**.** The presentation process helped me to understand better what kind of audiences would be interested in this version of DeltaVR and what kind of feedback to get from them. It also helped me to identify the potential problems that can arise during the testing.

Conclusion

I addressed several improvements and extensions of DeltaVR in this work which focus on enhancing its functionality, user engagement, and emotional impact. The key contributions include the development of a Digital Twin of the building, the integration of Emotional Goal Modeling to design immersive experiences, and the testing and presentation of the application for user feedback and refinement.

The first step in building a digital twin of the building was to integrate the weather data of Tartu. Section 4.2 mentions how weather is integrated using an API from OpenWeatherMap. A major part of this work involved integrating building data such as CO_2 levels, room temperatures, and overall energy consumption. Section 4.3 is dedicated to solving this problem. It contains detailed descriptions and steps (sections 4.3.1 and 4.3.2) on how to understand the data from the InfluxDB and use it in Unity, along with adding new dashboards and configuring URLs to obtain data for different rooms.

I have used Emotional Goal Modeling [7] to come up with experiences that can potentially be integrated along with the emotions they elicit in the users. To address the challenge of making the Delta VR app more engaging and attractive, various experiences from popular VR platforms such as VRChat, Rec Room, and Resonite were explored. The goal was to identify experiences (present in section 3.3) that would resonate well with Delta VR users by using Emotional Goal Modeling [7]. Emotional Goal Modeling is an approach in requirements engineering that emphasizes the importance of addressing emotional aspects as primary considerations in system design, rather than treating them as secondary or "non-functional" requirements.

I examined the diverse range of experiences offered by VRChat, Rec Room, and Resonite. These platforms are known for their detailed, immersive environments and interactive elements. By analyzing what makes these experiences captivating, the most suitable features and activities for Delta VR were identified. The focus was on experiences that not only align with Delta VR's objectives but also have the potential to elicit specific emotions. To assist future development in integrating these new experiences into Delta VR, instructions on implementing the features using Unity have been provided. This section includes step-by-step guidance on developing and the experiences.

The instructions are designed to be clear and practical, facilitating the effective implementation of the enhanced experiences. Along with this, applying emotional theories and engaging in user testing helped in gathering valuable insights into how users interact with and respond to different features. This process allowed for the potential refinement and enhancement of Delta VR's features based on real user feedback. I compiled a detailed table (see Fig. 4), outlining additional improvements and features along with their expected impact on user experience.

A significant part of this project involved the testing and presentation of Delta VR to assess usability and gather user feedback. This process serves as a guide for anyone looking to test or present the application to an audience. The initial usability testing covered participant selection, questionnaire design, and results analysis. Additionally, a second presentation session was conducted to demonstrate Delta VR to an audience, collect feedback, and analyze responses. This phase focused on identifying the right audience, structuring tailored questions for different tester groups, and recording feedback effectively. The results from this session were documented, helping to refine Delta VR based on user perceptions and suggestions..

Acknowledgment

I would like to express my gratitude to my supervisor, Ulrich Norbisrath, for his invaluable guidance and support throughout this project. I also extend my thanks to Raimond Tunnel for conducting the seminars and to Noffel Khan for his thorough peer review and insightful suggestions. Additionally, I appreciate Alicia Sudlerd for her valuable input and Pelle Jakovits for providing access to the building data. Lastly, I would like to thank all the students who participated in the testing sessions for their time and feedback.

I have used ChatGPT throughout the development of this thesis to refine and enhance the clarity and structure of the text. It has assisted me in editing technical descriptions, improving grammar, rephrasing content for better readability, and ensuring consistency across different sections. Additionally, I have utilized ChatGPT to summarize key concepts and simplify implementation steps. I have also used Grammarly to further enhance the grammar in the text.

Appendix

I. Dictionary

1. **Cybersickness**: This occurs when exposure to a virtual environment causes symptoms that are similar to motion sickness symptoms. The most common symptoms include general discomfort, eye strain, and headache
2. **FBX (Filmbox)**: A file format for the exchange and interoperability of 3D assets, animations, and scenes between various 3D modeling, animation, and rendering software.
3. **API key**: A unique identifier used to authenticate and authorize access to an API.
4. **Global-Configuration:** A centralized setup of settings or parameters that apply universally across an application, system, or environment to ensure consistency and simplify management.
5. **Light Baking** – A rendering process that precomputes lighting for better performance.
6. **Frame Time** – The time taken to render a single frame in a VR or gaming application.
7. **Ergodic Agency** – The ability of users to influence events and world changes over time.
8. **Congruence (Sensory)** – The alignment of multiple sensory inputs for realism in VR.
9. **Panoramic Experience** – A 360-degree immersive visual representation in VR.
10. **Shorthand** – A shorter or simplified way of writing or representing something. In data models, it refers to abbreviated names used for measurements or variables.
11. **Overlay** – A graphical element that appears on top of another interface or visual component. In VR or UI design, it refers to a layer that provides additional information or interaction without replacing the main content.
12. **Coroutine** – A function that can pause and resume execution, commonly used in programming (like in Unity) to handle asynchronous operations, such as making API calls without freezing the application.

    Pathfinding – A computational process used in AI and game development to determine the most efficient route from one point to another within a virtual space, often implemented using algorithms like A* or Unity's NavMesh.

13. **Immersion** – The feeling of being completely absorbed or surrounded by a virtual environment, making it feel real. In VR, this happens when users feel like they are truly inside the digital world rather than just looking at a screen.
14. **Nostalgia** – A sentimental longing or affection for the past, often triggered by familiar experiences, objects, or memories.
15. **Camaraderie** – A feeling of mutual trust and friendship between people who share a common activity or goal.

II. Steps to Run VPN on Oculus Quest

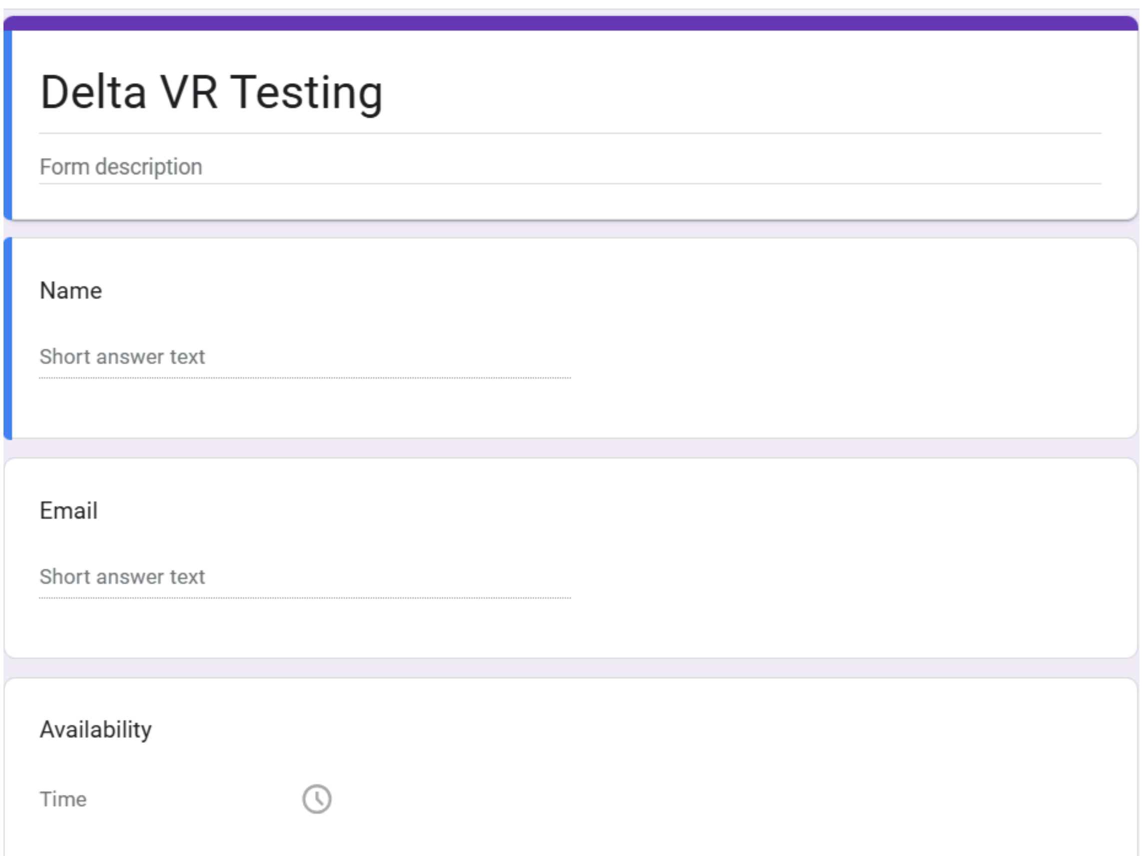
To access the building's data through the university's influxDB, a VPN must be used, and these below are the steps to install and run the VPN on the Meta Quest headset.

1. Download SideQuest: SideQuest is a platform that allows users to sideload apps and games onto the Oculus Quest. Start by downloading the SideQuest software from its official website here.
2. Install SideQuest: After downloading, install the SideQuest application on your computer. Follow the on-screen instructions to complete the installation process.
3. Connect Quest to Computer: Use a USB-C cable to connect your Oculus Quest headset to your computer. Ensure that your Quest is turned on.
4. Enable Developer Mode on Quest: To sideload apps, your Quest needs to be in Developer Mode. Open the Oculus app on your smartphone, go to Settings, select your Oculus Quest, tap More Settings, and then tap Developer Mode. Toggle Developer Mode on. If you haven't done so already, you may need to create an Oculus Developer account.
5. Open SideQuest and Connect Quest: Launch the SideQuest application on your computer. If your Quest is properly connected and Developer Mode is enabled, you should see a green dot in the top-left corner of SideQuest, indicating that your device is connected.
6. Add APK to SideQuest: In SideQuest, you can drag and drop the Global Protect APK file (downloaded from: https://apkpure.com/globalprotect/com.paloaltonetworks.globalprotect) into the SideQuest window or click the "Install APK file from folder on computer" button (an

icon with a folder and down arrow) to browse and select the APK file you want to install.

7. Check Installation Status: Once the APK is added, SideQuest will display the installation progress. You can check the status in the Tasks tab (a clipboard icon). Ensure the installation completes without errors.
8. Enable Unknown Sources on Quest: Put on your Oculus Quest headset. Navigate to the Library and look for the "Unknown Sources" category in the drop-down menu. Select it to see the sideloaded apps. This step ensures you can run apps that are not from the Oculus Store.
9. Follow the steps to establish a VPN connection: https://wiki.ut.ee/pages/viewpage.action?pageId=183223213

III. Testing and Presentation



Delta VR Testing

Form description

Name

Short answer text

Email

Short answer text

Availability

Time

Figure 24. The form used to invite the participants

Name:

Occupation/Program: Computer Science (MSc) || Computer Vision

- **Digital Twin Issues & Improvement**
  - General opinion about the VR based digital twin

Overall liked the whole experience

The user needs to know what to do and where to go.

Users are able to find dashboards

  - Is there any other way the Building Data can be presented - Dashboards?
- **UI Improvement**
  - Is it easy & understandable (Check that yourself too)
  - How can the UI be improve
- **Experience Ideas**
  - These are some of the experiences, what can be added more to it?

Give teleport options for different places

Mention the place where you are right now (For outsiders)

Figure 25. Feedback form and a student testing the application during the second presentation

References

1. Tamm, T. 2023. Delta VR: Multiplayer 2.0 [University of Tartu]. https://dspace.ut.ee/server/api/core/bitstreams/4d6d0190-9004-4ed9-9978-b456a04fba25/content
2. Haiqin Xie, Lin XIAO*, Xinguo Ming, Sheng Tan, and Yuguang Bao. 2023. Driving Intelligent Manufacturing: An Application Study on Digital Twin in Factory Digitalization. In 2023, The 10th International Conference on Industrial Engineering and Applications. https://doi.org/10.1145/3587889.3588216
3. Jens-Stefan Mikson. 2017. Virtual Reality Game Design Analysis Based on Tribocalypse VR. https://comserv.cs.ut.ee/ati_thesis/datasheet.php?id=58201
4. Bodi, Bettina and Thon, Jan-Noël. "Playing stories?: Narrative-dramatic agency in *Disco Elysium* (2019) and *Astroneer* (2019)" *Frontiers of Narrative Studies*, vol. 6, no. 2, 2020, pp. 157-190. https://doi.org/10.1515/fns-2020-0012
5. Kauhanen, O., Väätäjä, H., Turunen, M., Keskinen, T., Sirkkunen, E., Uskali, T., Lindqvist, V., Kelling, C., & Karhu, J. (2017). Assisting Immersive Virtual Reality Development with User Experience Design Approach. In AcademicMindtrek'17, September 20–21, 2017, Tampere, Finland. https://doi.org/10.1145/3131085.3131126
6. Jerald, J. (2015). The VR Book: Human-Centered Design for Virtual Reality. Morgan & Claypool. https://doi.org/10.1145/2792790
7. Tahira Iqbal, James George Marshall, Kuldar Taveter, Albrecht Schmidt, Theory of constructed emotion meets RE: An industrial case study, Journal of Systems and Software, Volume 197, 2023, 111544, ISSN 0164-1212, https://doi.org/10.1016/j.jss.2022.111544.
8. T. Pexyean, K. Saraubon, and P. Nilsook, "IoT, AI and Digital Twin For Smart Campus," 2022 Research, Invention, and Innovation Congress: Innovative Electricals and Electronics (RI2C), Bangkok, Thailand, 2022, pp. 160-164, doi: 10.1109/RI2C56397.2022.9910286.
9. N. A. Fashal, G. A. Elkhayat and S. A. Elmorsy, "Review and Proposed Digital Twin Model for Sustainable Smart University Asset Management," 2023 IEEE

Afro-Mediterranean Conference on Artificial Intelligence (AMCAI), Hammamet, Tunisia, 2023, pp. 1-8, doi: 10.1109/AMCAI59331.2023.10431490.

10. Tamm, T. 2021. Delta VR [University of Tartu] https://comserv.cs.ut.ee/ati_thesis/datasheet.php?id=71682
11. Püks, J. (2022). DeltaVR - Multiplayer [University of Tartu]. https://dspace.ut.ee/server/api/core/bitstreams/bbcb520b-025a-4bdb-b15a-cf76cf7d4752/content

License

Non-exclusive licence to reproduce the thesis and make the thesis public

I, Muhammad Taimour Khan,

1. Grant the University of Tartu a free permit (non-exclusive licence) to reproduce, for preservation, including for adding to the DSpace digital archives, until the expiry of the term of copyright, my thesis

DeltaVR: Digital Twin

Supervised by Ulrich Norbisrath

2. I grant the University of Tartu a permit to make the thesis specified in point 1 available to the public via the web environment of the University of Tartu, including via the DSpace digital archives, under the Creative Commons licence CC BY NC ND 4.0, which allows, by giving appropriate credit to the author, to reproduce, distribute the work and communicate it to the public, and prohibits the creation of derivative works and any commercial use of the work until the expiry of the term of copyright.

3. I am aware of the fact that the author retains the rights specified in points 1 and 2.

4. I confirm that granting the non-exclusive licence does not infringe other persons' intellectual property rights or rights arising from the personal data protection legislation.

Muhammad Taimour Khan
14/05/2025